

3100000011475



REKAM

INSTRUKSI

NO

3961

**TUGAS AKHIR (NA. 1701)**

**KOMPUTERISASI PENYIMPANGAN BENTUK  
KOMPONEN 3-D KAPAL DALAM PROSES PRODUKSI  
DENGAN METODE *MATCHING PROCESS***



RSPe  
623.820 028 5  
Kri  
k-1  
1998

6-a-gg  
H  
8433

**OLEH :**

**ANTON KRISTANTO**  
**NRP. 4192 100 016**

**JURUSAN TEKNIK PERKAPALAN  
FAKULTAS TEKNOLOGI KELAUTAN  
INSTITUT TEKNOLOGI SEPULUH NOPEMBER SURABAYA**

**1998**



# JURUSAN TEKNIK PERKAPALAN

## FAKULTAS TEKNOLOGI KELAUTAN ITS

### SURAT KEPUTUSAN TUGAS AKHIR (NA 1701)

No. : 41 /PT12.FTK2/M/1997

Nama Mahasiswa : Anton Kristanto  
Nomor Pokok : 4192100016  
Tanggal diberikan tugas : 16 Maret 1997  
Tanggal selesai tugas : 26 Juli 1997  
Dosen Pembimbing : 1. Ir. Sjarief Widjaja, Ph.D.  
2. ....

#### Uraian / judul tugas akhir yang diberikan :

~~KOMPUTERISASI PENYIMPANGAN BENTUK KOMPONEN 3-D KAPAL DALAM PROSES PRODUKSI-~~  
~~DENGAN METODE MATCHING PROCESS-~~

sOn

Surabaya, 31 Maret 1997  
Jurusan Teknik Perkapalan FTK-ITS



#### Tembusan :

1. Yth. Dekan FTK-ITS.
2. Yth. Dosen Pembimbing.
3. Arsip.

Dr. Mestoro Sastro Wiyono  
NIP. 130.687.430.

**LEMBAR PENGESAHAN**

**TUGAS AKHIR (NA. 1701)**

**KOMPUTERISASI PENYIMPANGAN BENTUK  
KOMPONEN 3-D KAPAL DALAM PROSES PRODUKSI  
DENGAN METODE *MATCHING PROCESS***

Oleh :

**ANTON KRISTANTO**  
**NRP. 4192 100 016**

Telah diperiksa dan dinyatakan siap untuk diujikan  
pada tanggal , 5 Maret 1998

Surabaya, 24 Pebruari 1998  
Dosen Pembimbing



**DR. Ir. SJARIEF WIDJAJA**

**LEMBAR PENGESAHAN**  
**TELAH DIREVISI SESUAI DENGAN PROSES VERBAL**

Surabaya, 16 Maret 1998  
Dosen Pembimbing



**DR. Ir. SJARIEF WIDJAJA**



## ABSTRAK

Dalam Tugas Akhir ini akan dibahas mengenai penyimpangan bentuk komponen 3-D yang terjadi dengan metode *matching process* pada proses produksi komponen bangunan kapal. *Matching process* yaitu pencocokan antara gambar 3-D sebenarnya berasal dari pengukuran suatu komponen hasil proses produksi dengan gambar 3-D perencanaan. Hasil yang diperoleh berupa data-data penyimpangan bentuk komponen bangunan kapal. Selanjutnya akan direncanakan suatu program dengan menggunakan komputer dalam melaksanakan metode *matching process*.

Metode *matching process* diperlukan sehubungan dengan masih banyaknya pekerjaan *rework* yang terjadi selama proses produksi komponen bangunan kapal yang dapat menambah biaya dan waktu produksi akibat penyimpangan bentuk dan dimensi. Pelaksanaan *matching process* dibutuhkan untuk mengetahui ketepatan ukuran dan bentuk komponen 3-D kapal guna menghindari pekerjaan *rework* dalam pembuatan komponen yang sejenis berikutnya.

Dengan menggunakan sistem komputerisasi diharapkan pelaksanaan metode *matching process* akan menjamin hasil yang memiliki akurasi tinggi.



## KATA PENGANTAR

Puji syukur kehadirat Allah SWT dengan karunia dan hidayah-Nya, sehingga penulis dapat menyelesaikan penyusunan Tugas Akhir ini. Tujuan dari penyusunan Tugas Akhir ini adalah untuk melengkapi persyaratan guna mencapai gelar Kesarjanaan S1 di Jurusan Teknik Perkapalan, Fakultas Teknologi Kelautan, Institut Teknologi Sepuluh Nopember Surabaya.

Adapun permasalahan yang dibahas dalam Tugas Akhir ini adalah :  
“Komputerisasi Penyimpangan Bentuk Komponen 3-D Kapal Dalam Proses Produksi Dengan Metode *Matching Process*” . Dalam penyusunan Tugas Akhir ini hingga selesai tak lepas dari dukungan dan bantuan dari berbagai pihak. Oleh karena itu penulis mengucapkan terima kasih yang sebesar-besarnya kepada :

1. Bapak Ir. Koestowo S.W., selaku Ketua Jurusan Teknik Perkapalan, Fakultas Teknologi Kelautan ITS Surabaya.
2. Bapak DR. Ir. Sjarief Widjaja, selaku Dosen Pembimbing yang membantu dan membimbing penyusunan Tugas Akhir ini.
3. Bapak Pimpinan serta segenap staf dan karyawan PT. (Persero) PAL INDONESIA Surabaya.
4. Bapak Pimpinan serta segenap staf dan karyawan Laboratorium Hidridinamika Indonesia (LHI).
5. Seluruh Dosen serta staff dan karyawan di Jurusan Teknik Perkapalan, Fakultas Teknologi Kelautan ITS Surabaya.

6. Kedua orang tua, Dede dan Rizqi, serta seluruh keluarga yang telah banyak membantu dan memberikan dorongan semangat.
7. Dan semua pihak baik secara langsung maupun tidak langsung turut mendukung terselesaikannya penyusunan Tugas Akhir ini yang tidak dapat penulis sebutkan satu persatu.

Dalam penyusunan Tugas Akhir ini penulis telah berusaha semaksimal mungkin, namun tidak lepas dari kekurangan dan kesalahan. Untuk itu saran dan kritik yang membangun sangat penulis harapkan. Akhir kata penulis berharap semoga Tugas Akhir ini dapat bermanfaat dan menambah wawasan ilmu pengetahuan bagi pembaca.

Surabaya, Pebruari 1998

Penyusun

## DAFTAR ISI

ABSTRAK	i
KATA PENGANTAR	ii
DAFTAR ISI	iv
DAFTAR GAMBAR	vii
DAFTAR TABEL	ix
BAB I    PENDAHULUAN	1
I.1.    LATAR BELAKANG MASALAH	3
I.2.    PERUMUSAN MASALAH	3
I.3.    BATASAN MASALAH	3
I.4.    TUJUAN DAN MANFAAT	3
I.4.1.    TUJUAN	3
I.4.2.    MANFAAT	3
I.5.    METODOLOGI DAN MODEL ANALISIS	4
I.6.    KESIMPULAN AWAL	4
BAB II    TEKNOLOGI PRODUKSI KAPAL	
II.1.    PEMBANGUNAN TEKNOLOGI PEMBANGUNAN KAPAL	5
II.2.    FAKTOR-FAKTOR PROSES PRODUKSI	9
II.3.    TAHAPAN PROSES PRODUKSI PEMBANGUNAN KAPAL	12
II.3.1.    TAHAP PERSIAPAN	12



II.3.2.	TAHAP <i>FABRIKASI</i>	13
II.3.3.	TAHAP <i>SUB ASSEMBLY</i>	18
II.2.5.	TAHAP <i>ASSEMBLY</i>	18
II.2.6.	TAHAP <i>ERECTION</i>	19
II.4.	PERMASALAHAN DALAM PROSES PRODUKSI	21
BAB III	PROSES PEMERIKSAAN BENTUK DAN DIMENSI	29
III.1.	PEMERIKSAAN PADA TAHAP <i>FABRIKASI</i>	29
III.2.	PEMERIKSAAN PADA TAHAP <i>SUB ASSEMBLY</i>	32
III.3.	PEMERIKSAAN PADA TAHAP <i>ASSEMBLY</i>	33
III.4.	PEMERIKSAAN PADA TAHAP <i>ERECTION</i>	34
BAB IV	KONSEP DASAR KOMPUTER GRAFIS	36
IV.1.	KONSEP DASAR BENDA 3-D	36
IV.2.	PENYAJIAN OBYEK 3-D	36
IV.2.1.	PERMUKAAN POLIGON ( <i>POLYGON SURFACES</i> )	36
IV.2.2.	JARING-JARING POLIGON ( <i>MESH</i> )	39
IV.2.3.	SISTEM KOORDINAT <i>CARTESIAN</i> 3-D	40
IV.3.	<i>COMPUTER AIDED DESIGN (CAD)</i>	42
BAB V	PEMBUATAN PROTOTIPE <i>MATCHING PROCESS</i>	44
V.1.	BENDA KERJA SEBAGAI PROTOTIPE	44
V.2.	SISTEM KOORDINAT <i>MATCHING PROCESS</i>	47
V.3.	<i>3D POLYGON MESHES</i>	48
V.4.	LANGKAH <i>MATCHING PROCESS</i>	49

V.5.	BAHASA PEMROGRAMAN <i>MATCHING PROCESS</i>	54
V.5.1.	<i>AutoLISP</i>	55
V.5.2.	<i>Visual BASIC</i>	56
BAB VI	PROTOTYPE PROGRAM <i>MATCHING PROCESS</i>	58
VI.1.	<i>FLOWCHART PROGRAM MATCHING PROCESS</i>	58
VI.2.	STRUKTUR PROGRAM <i>MATCHING PROCESS</i>	59
V.2.1.	MAIN.FRM	60
V.2.2.	HASIL.FRM	61
V.2.3.	KURVAMX.FRM, KURVAMY.FRM, KURVAMZ.FRM	62
V.2.4.	KURVANX.FRM, KURVANY.FRM, KURVANZ.FRM	63
V.2.5.	KTR.FRM	63
VI.3.	<i>RUNNING PROGRAM</i>	64
BAB VII	DISKUSI DAN REKOMENDASI	72
BAB VIII	KESIMPULAN	75
DAFTAR PUSTAKA		x
LAMPIRAN		

## DAFTAR GAMBAR

Gambar II-1	Tahapan Perkembangan Teknologi Produksi	5
Gambar II-2	Komponen <i>Product Oriented Work Breakdown System</i>	7
Gambar II-3	Urutan proses produksi pada bengkel fabrikasi	15
Gambar IV-1	Gambar dua permukaan yang saling bersinggungan	38
Gambar IV-2	Jaring segitiga yang dibentuk oleh 11 segitiga menghubungkan 13 <i>vertex</i>	39
Gambar IV-3	Jaring empat sisi mengandung 12 segiempat yang dibentuk Dari 5 x 4 <i>array vertex</i> input	40
Gambar IV-4	Sistem koordinat tiga dimensi	41
Gambar IV-5	Koordinat <i>Cartesian</i> dengan kaidah tangan kanan	41
Gambar V-1	Penandaan bilga gading no. 84 – 93 PAX 500	45
Gambar V-2	Penentuan titik origin koordinat 3-D <i>Matching Process</i>	47
Gambar V-3	Arah M dan N pada pelat kulit	48
Gambar V-4	Langkah <i>Matching Process</i>	50
Gambar V-5	Penggambaran komponen 3-D desain dan aktual	52
Gambar V-6	Penggambaran proses <i>Matching</i>	52
Gambar VI-1	<i>Flowchart</i> program <i>Matching Process</i>	58
Gambar VI-2	Tampilan <i>Main.Frm</i>	60
Gambar VI-3	Tampilan <i>Hasil.Frm</i>	62
Gambar VI-4	Tampilan <i>Ktr.Frm</i>	63
Gambar VI-5	Penggambaran grafis komponen 3-D	67
Gambar VI-6	Grafik Penyimpangan X Potongan M	69

Gambar VI-7	Grafik Penyimpangan Y Potongan M	69
Gambar VI-8	Grafik Penyimpangan Z Potongan M	70
Gambar VI-9	Grafik Penyimpangan X Potongan N	70
Gambar VI-10	Grafik Penyimpangan Y Potongan N	71
Gambar VI-11	Grafik Penyimpangan Z Potongan N	71





## DAFTAR TABEL

Tabel III-1	Urutan pemeriksaan tiga tahap	30
Tabel IV-1	Data geometri yang menyajikan dua permukaan poligon	38
Tabel VI-1	Data Desain (Titik Koordinat 3-D)	65
Table VI-2	Data Aktual (Titik Koordinat 3-D)	66
Tabel VI-3	Data Penyimpangan	68

**BAB I**  
**PENDAHULUAN**



# **BAB I**

## **PENDAHULUAN**

### **I.1. LATAR BELAKANG MASALAH**

Dewasa ini galangan kapal Nasional mulai menerapkan sistem *Accuracy Control (A/C)* pada proses produksi guna memenuhi tuntutan konsumen mengenai harga murah, waktu pembuatan yang singkat, dan hal yang penting adalah mutu bangunan kapal yang sesuai dengan persyaratan yang ditetapkan. Sistem ini diterapkan agar galangan Nasional dapat bersaing di pasar dunia dalam bidang industri perkapalan.

Maksud diterapkannya sistem *A/C* yaitu untuk mencapai suatu standar mutu galangan yang digunakan sebagai pengendali mutu produk sekaligus biaya produksi. Sistem *A/C* yang digunakan merupakan salah satu bentuk pengembangan teknologi produksi yang diarahkan untuk mengetahui penyimpangan bentuk dan dimensi komponen badan kapal selama proses produksi.

Dalam melaksanakan tiap proses produksi pembuatan komponen kapal selalu berdasarkan gambar kerja dari bagian perencanaan. Komponen yang dihasilkan selama proses produksi belum tentu mempunyai bentuk dan dimensi yang benar-benar sesuai dengan rencana pada gambar kerja. Penyimpangan bentuk dan dimensi yang terjadi tidak dapat diidentifikasi pada tahap dini, tetapi penyimpangan yang terjadi baru dapat diketahui pada tahap lanjut (proses *erection*), dimana penyelesaian sudah cukup sulit.

Pada saat ini untuk mengetahui kesalahan yang mungkin terjadi selama proses produksi digunakan metode pemeriksaan yang merupakan perwujudan konsep A/C. Pemeriksaan ini dilaksanakan untuk setiap langkah pekerjaan pada tiap tahapan proses produksi.

Urutan pemeriksaan selalu dilaksanakan oleh pekerja sampai *surveyor* QA/C maupun *owner* berdasarkan standar atau ketentuan yang telah disepakati bersama. Untuk mengidentifikasi kesalahan-kesalahan tersebut sangat tergantung dari keahlian dan pengalaman pelaku pemeriksa. Isi data laporan yang berhubungan dengan kesalahan penyimpangan bentuk dan dimensi hanya bersifat mengidentifikasi, tetapi belum memberikan informasi besar penyimpangan secara mendetail dan akurat. Hal ini dikarenakan belum adanya suatu metode yang dapat digunakan untuk mengukur besar penyimpangan tersebut secara detail dan akurat.

Oleh karena itu penulis menganggap perlu dicari suatu metode yang dapat digunakan. Metode yang dicari harus dapat menghasilkan suatu angka yang menyebutkan besar penyimpangan bentuk dan dimensi dengan mendetail dan akurat. Metode ini dinamakan *matching process* yaitu pencocokkan antara data dari pengukuran komponen 3-D hasil proses produksi terhadap detail ukuran dan detail dari gambar perencanaan komponen kapal. Dengan mencocokkan kedua data ini diharapkan bahwa besar penyimpangan bentuk dan dimensi dapat didapatkan. Pelaksanaan metode ini menggunakan program komputer, sehingga data yang dihasilkan detail dan akurat dan tidak memerlukan orang dengan keahlian khusus untuk dapat mengoperasikannya.



## I.2. PERUMUSAN MASALAH

Bagaimana penyusunan metode *matching process* dengan menggunakan pemrograman komputer agar besar penyimpangan bentuk komponen 3-D kapal dapat diketahui.

## I.3. BATASAN MASALAH

Batasan permasalahan Tugas Akhir ini adalah pada konsep dasar *matching process* mengenai langkah atau logika yang digunakan.

## I.4. TUJUAN DAN MANFAAT

### I.4.1. TUJUAN

- Memberikan informasi dalam bentuk data penyimpangan bentuk komponen 3-D kapal dalam proses produksi secara detail dan akurat.
- Penggunaan sistem komputerisasi data yang diperoleh mempunyai tingkat akurasi yang tinggi dan proses dapat dilakukan dengan cepat.

### I.4.2. MANFAAT

- Mempercepat pelaksanaan pemeriksaan komponen kapal pada semua tahapan proses produksi.
- Menghindari pekerjaan *rework* dalam proses produksi suatu komponen kapal apabila kesalahan penyimpangan bentuk dan dimensi dapat diketahui secara cepat, detail dan akurat.

## I.5. METODOLOGI DAN MODEL ANALISIS

- *Studi Literatur*

Penulisan tugas akhir berdasarkan literatur-literatur yang mempunyai relevansi dengan permasalahan.

- *Program Komputer*

Permasalahan dianalisa dengan pendekatan numerik sehingga mudah untuk dimanipulasi dengan komputer.

- *Analisa Hasil*

Menganalisa hasil dari program yang dibuat.

## I.6. KESIMPULAN AWAL

Dengan melaksanakan metode *matching process* dengan sistem komputerisasi identifikasi penyimpangan bentuk komponen 3-D kapal pada proses produksi dapat diperoleh dengan tingkat ketelitian yang baik.





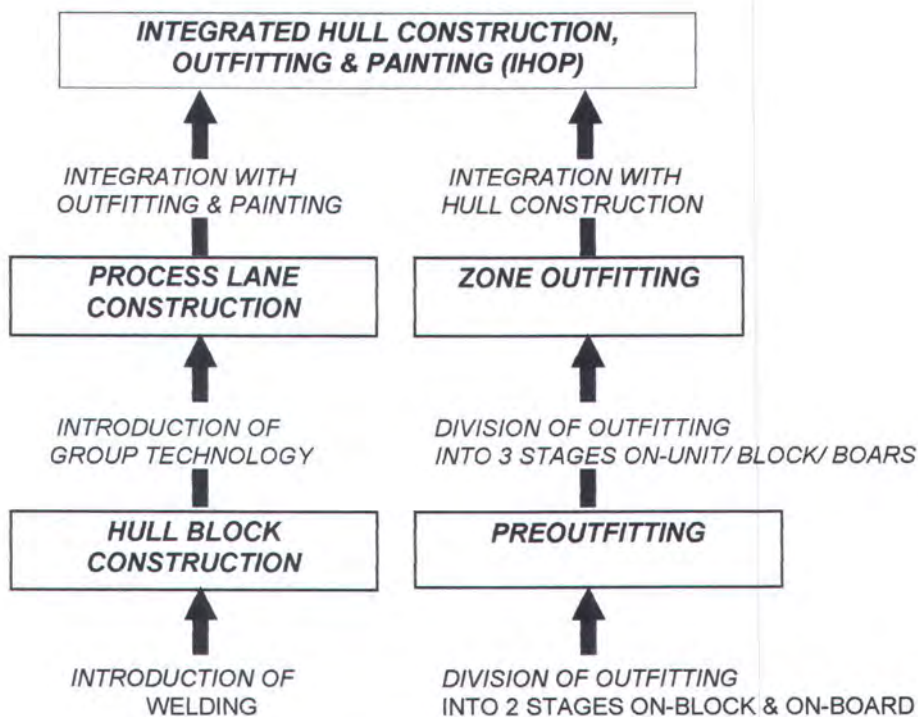
**BAB II**  
**TEKNOLOGI PRODUKSI**  
**KAPAL**

## BAB II

### TEKNOLOGI PEMBANGUNAN KAPAL

#### II.1. PERKEMBANGAN TEKNOLOGI PEMBANGUNAN KAPAL

Teknologi pembangunan kapal mulai berkembang pesat setelah Perang Dunia II, yang disponsori oleh IHI-Shipyards, Japan. Pengembangan tersebut didasarkan pada usaha peningkatan efisiensi pada proses produksi atau produktivitas pada pembangunan beberapa kapal Tanker berukuran besar. Menurut *Chirillo*, perkembangan teknologi produksi kapal dibagi menjadi *empat tahapan*, berdasarkan teknologi yang digunakan pada proses produksinya, seperti pada *Gambar II-1* berikut :



**Gambar II-1**  
Tahapan Perkembangan Teknologi Produksi



#### ❑ ***Conventional Construction and Outfitting***

Tahapan pertama ini merupakan teknologi produksi kapal, yang berorientasi pada *sistem atau fungsi* yang ada di kapal dan volume pekerjaan hampir seluruhnya dilakukan pada building berth. Metode ini dimulai dengan peletakan lunas, kemudian pemasangan gading, kulit. Dst sampai ke bangunan atas dan terakhir pekerjaan outfitting (O/F). Pekerjaan O/F dilakukan sistem per sistem, antara lain : pemasangan ventilasi, perpipaan, permesinan, perlistrikan, dll. Metode tersebut merupakan teknologi paling *konvensional* dan tingkat produktivitasnya sangat rendah, karena semua lingkup pekerjaan memiliki ketergantungan yang tinggi satu sama lain, sehingga membutuhkan waktu yang sangat lama. Selain itu, mutu hasil pekerjaan sangat rendah karena hampir seluruh pekerjaan dilakukan secara manual dan lingkungan kerja yang tidak mendukung (tidak nyaman, posisi kerja yang sulit).

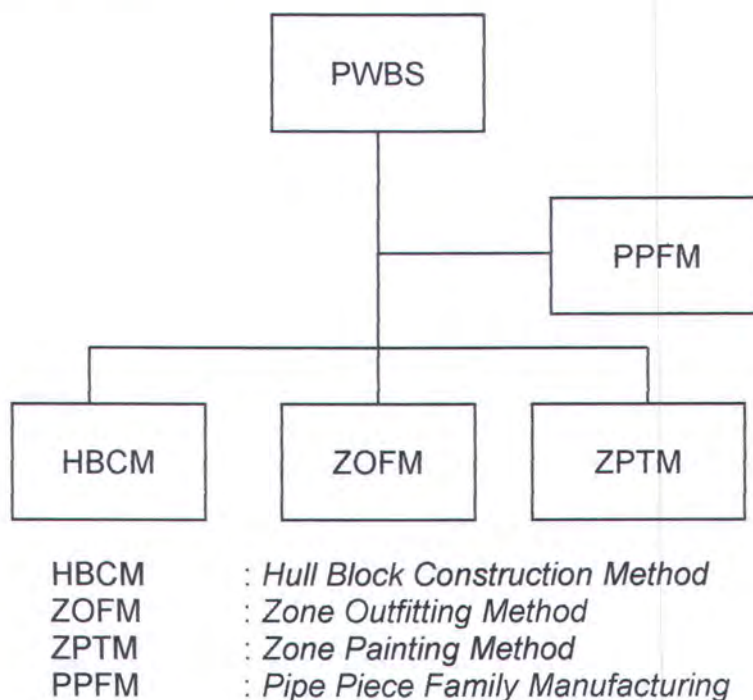
#### ❑ ***Hull Block Construction Method & Pre-Outfitting***

Tahapan ini mulai dikembangkan sejak dikenalkannya *teknologi pengelasan* pada pembangunan kapal, dimana kapal sudah dibuat dalam bentuk seksi-seksi dan blok-blok, kemudian disambung satu sama lain di building berth. Selain itu, beberapa pekerjaan O/F sudah mulai dilakukan pada blok atau badan kapal yang sudah jadi. Dengan metode ini, *steel throughput* meningkat dan mutu hasil pekerjaan lebih baik, karena volume pekerjaan pada building berth menjadi berkurang, dan pekerjaan pengelasan banyak dilakukan di bengkel dengan kondisi lingkungan kerja yang lebih nyaman. Pekerjaan pengelasan sudah mulai dilakukan dengan *mesin las semi/otomatis* dengan posisi *down-hand*. Blok-blok dapat dibalik atau diputar untuk menghindari dari pengelasan dengan posisi

*overhead*. Pada tahapan ini, *kontrol dimensi* dan *bentuk blok* sudah menjadi penting terutama pada daerah sambungan blok.

#### ❑ **Process Lane Construction and Zone Outfitting**

Perkembangan teknologi pembangunan kapal modern telah dimulai pada tahap ini, dimana konsep *Group Technology* sudah diterapkan dalam proses produksi badan kapal dan pekerjaan O/F. Dengan konsep ini proses pembangunan kapal sudah berorientasi pada produk atau dikenal dengan *Product-Oriented Work Breakdown Structure* (PWBS), dengan pengelompokan lingkup pekerjaan seperti terlihat pada *Gambar II-2*.



**Gambar II-2**  
Komponen *Product-Oriented Work Breakdown Structure*

Metode ini secara sistematis mengklasifikasikan *produk antara* (*interim-products*) menurut kelompok yang memiliki kesamaan dalam proses produksinya,



misalnya *process lane* untuk : (i) perakitan bentuk datar, (ii) perakitan bentuk lengkung beraturan, dan (iii) perakitan bentuk lengkung tak beraturan dan kompleks. Hal ini sudah menuntut keteraturan dalam penataan sumberdaya produksi, misalnya : peralatan/mesin, tenaga kerja, dan material. Demikian halnya dengan pekerjaan O/F, dimana sudah dilakukan secara paralel berdasarkan *region/zone*, dan tidak lagi berdasarkan sistem fungsionalnya. Pekerjaan O/F, dengan konsep *Advanced Outfitting*, sudah dibagi dalam 3 tahap yaitu : *on-unit*, *on-block*, dan *on-board*.

Pada tahapan ini, proses produksi sudah terkonsentrasi pada bengkel-bengkel dan volume pekerjaan di *building berth* semakin kecil. Penggunaan teknologi ini sudah mensyaratkan diterapkannya *Sistem Accuracy Control*, di setiap proses produksi untuk menjamin *ketepatan dimensi* dan *bentuk blok* (pelat, profile, perpipaan, dan *inner-parts* lainnya yang ada pada daerah sambungan blok).

#### ❑ ***Integrated Hull Construction, Outfitting, and Planning (IHOP)***

Tahapan keempat ini adalah teknologi yang paling *mutakhir* dalam pembangunan kapal, dimana proses pembuatan badan kapal sudah *diintegrasikan* secara maksimal dengan pekerjaan *outfitting* dan *pengecatan* pada setiap *zone/area/stage*, seperti halnya yang telah dilakukan oleh IHI-Shipyards, Japan. Teknologi ini merupakan pengembangan dari teknologi produksi tahapan ketiga, melalui peningkatan potensi sumberdaya galangan secara *menyeluruh*, *seimbang* dan *terintegrasi*, disertai dengan semakin sempurnanya standar-standar kerja dan konsistensi ketepatan proses produksi yang sangat tinggi.



Teknologi IHOP mensyaratkan suatu *Build Strategy* yang matang dan kemampuan *planning and schedulling* yang sangat tinggi dan rasionil, berdasarkan kondisi potensi sumberdaya galangan yang ada. Pada tahap ini, Sistem Accuracy Control bukan hanya mutlak dilaksanakan secara sempurna dan menyeluruh, tetapi juga menuntut kesempurnaan *design engineering* dan *standar-standar* kerjanya. Dengan demikian, penyambungan seluruh bagian konstruksi antara blok yang satu dengan blok lainnya mencapai tingkat ketepatan yang sangat tinggi atau penyimpangan-penyimpangan yang terjadi masih berada pada batas toleransi yang telah ditentukan, sehingga volume pekerjaan pada building berth menjadi sangat kecil.

## II.2. FAKTOR-FAKTOR PROSES PRODUKSI

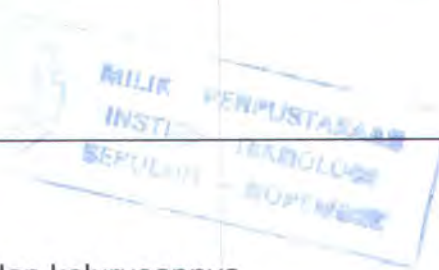
Pada proses pembangunan kapal faktor-faktor yang mempengaruhi proses produksi adalah :

### □ **Material**

Pada pembangunan kapal material yang dibutuhkan sangat mempengaruhi jalannya proses produksi. Untuk material logam baja diperlukan proses pengerjaan meliputi pemotongan, pembentukan, penggabungan pelat dan profil. Berdasarkan proses pengerjaan dari masing-masing material yang berbeda maka peralatan yang dibutuhkan, kualifikasi tenaga kerja, tata letak bengkel dan prosedur kerja yang digunakan akan sangat berbeda. Sehingga ada beberapa hal yang harus diperhatikan di dalam pemakaian material yaitu :

#### ▪ Spesifikasi Teknik Material

Meliputi : sifat-sifat mekanik, seperti kekuatan tarik, tekan, kemuluran dan beberapa sifat kimia kandungan logamnya.



- Ukuran/dimensi

Meliputi : ketebalan, panjang, lebar dan kelurusannya.

- Pabrik pembuat/produsen

Meliputi : kualifikasi dari pabrik pembuat dilihat dari cacatan mutu produk yang dihasilkan.

- Lingkungan/cuaca/temperatur/kelembaban.

Yaitu merupakan media dimana material tersebut akan dikerjakan atau diproses. Hal ini tergantung dari tempat proses pekerjaan dilakukan.

Misalnya untuk di Indonesia maka kondisi lingkungannya akan sangat berbeda dengan Eropa.

#### □ Tenaga Kerja

Kualifikasi tenaga kerja pada proses produksi akan sangat berbeda, tergantung dari jenis proses pengerjaan juga jenis material yang akan dikerjakan. Disamping itu jumlah tenaga kerja yang ada juga bervariasi sesuai dengan kebutuhan pada masing-masing tahap proses produksi. Sedangkan untuk pemakaian tenaga kerja dapat diklasifikasikan dari segi :

- Pendidikan formal/non formal

Meliputi latar belakang pendidikan seperti STM, Politeknik, dan kualifikasi khusus seperti *Welder Qualification grade G3*, dll. Hal ini sangat berpengaruh terhadap pemilihan tenaga kerja.

- Pengalaman dan masa kerja

Seorang tenaga kerja yang telah mempunyai pengalaman dalam bidang yang sesuai akan sangat membantu pada proses pelaksanaan pekerjaan.



- **Ketrampilan/Skill (kualifikasi)**

Ketrampilan khusus yang dimiliki oleh tenaga kerja akan sangat membantu didalam proses pekerjaan. Misalnya pada teknologi pengelasan logam tipis, membending pelat, dan sebagainya.

- **Sikap/Karakter**

Sikap dan karakter setiap tenaga kerja akan sangat membantu pada pelaksanaan proses produksi sehingga akan menciptakan iklim kerja sesuai dengan yang diinginkan.

- **Metode Produksi**

Hal-hal yang harus diperhatikan pada penetapan metode produksi adalah yang berkaitan langsung dengan tugas dari masing-masing tenaga kerja, diantaranya :

- Standar dan Prosedur kerja (*Operating Procedure Agreement*)
- Urutan pekerjaan (*Operating Instruction*)
- Peralatan/perlengkapan keselamatan kerja

- **Peralatan Produksi**

Pada penerapan peralatan kerja maka kita harus memperhatikan hal-hal sebagai berikut :

- Spesifikasi teknik
- Kondisi Operasional
- Kalibrasi dan Sistim Pemeliharaan
- Masa Operasi dan Penyusutan



## II.3. TAHAPAN PROSES PRODUKSI PEMBANGUNAN KAPAL

Pada setiap pembangunan kapal baru selalu melalui proses produksi. Proses pembangunan kapal itu sendiri dimulai sejak material datang sampai dengan penyerahan kapal kepada pihak pemesan (*delivery*). Penerapan proses produksi kapal meliputi beberapa tahap diantaranya yaitu : tahap persiapan produksi (yang meliputi perancangan dan persiapan gambar kerja, penyimpanan dan pemeriksaan material, persiapan tenaga kerja dan material, dll), tahap *fabrikasi* (pemotongan dan pembentukan material), tahap *sub-assembly* (penggabungan beberapa komponen dasar), tahap *assembly* (penggabungan beberapa komponen dasar menjadi blok utama), tahap *erection* (penggabungan blok-blok utama menjadi kapal secara utuh).

Pada masing-masing tahapan proses produksi selama pelaksanaannya masih banyak terjadi penyimpangan hasil produksi sehingga akan menghambat jalannya proses produksi karena waktu yang diperlukan untuk pembangunan kapal akan bertambah (*rework*) dan tentu saja hal ini akan menambah biaya produksi.

### II.3.1. TAHAP PERSIAPAN

Pada galangan yang akan melaksanakan pembangunan kapal, tahap pertama yang harus dilaksanakan adalah tahap persiapan produksi. Dalam tahap ini mempunyai tujuan mengatur keadaan-keadaan sehingga pada waktu yang ditentukan pekerjaan pembangunan kapal dapat dilaksanakan dan ditetapkan. Pada tahap ini ruang lingkup yang dikerjakan adalah :

- Dokumen-dokumen produksi (umum) ; meliputi gambar dan daftar material, perkiraan kebutuhan tenaga kerja, perkiraan kebutuhan material, dan jumlah

tenaga kerja yang terlibat dalam kaitannya dengan kapasitas tenaga kerja dan pekerjaan yang lain.

- Penyediaan material dengan mempertimbangkan keadaan atau stock pada gudang, pemakaian material untuk pekerjaan sekarang, dan pemesanan ataupun pembelian material dari luar (jumlah, waktu pemberian).
- Kapasitas dari sarana-sarana produksi meliputi kemampuan bengkel-bengkel produksi, kapasitas mesin-mesin, alat-alat angkat yang tersedia (jumlah, kapasitas, macam dan tempat), dan keadaan *building berth/floating dock* (jumlah, kapasitas dan macam).

### II.3.2. TAHAP *FABRIKASI*

Pada tahap ini dilaksanakan pembuatan komponen kapal dari yang kecil hingga yang terbesar. Jadi bengkel ini merupakan bengkel awal bagi pembuatan kapal secara fisik.

Dalam pelaksanaan proses pekerjaan pada tahap ini selalu diperlukan gambar-gambar dan rambu sebagai pedoman pemrosesan komponen. Gambar-gambar dan rambu-rambu yang diperlukan adalah sebagai berikut :

- *Marking List*

Untuk mengetahui bentuk dari komponen-komponen yang akan dikerjakan dalam satu blok. *Marking List* ini memuat antara lain : nomor kapal, nomor blok, serta ukuran-ukuran dan tanda-tandanya.

- *Material List*

Untuk mengetahui jumlah dari komponen-komponen yang akan dikerjakan dalam satu blok, berat blok tersebut, dan tempat komponen-komponen



tersebut dikerjakan. Selain itu juga untuk mengecek komponen-komponen yang belum dikerjakan.

- *Cutting Plan*

Untuk mengetahui jumlah dari material-material yang dibutuhkan dalam satu blok baik berupa pelat, profil, bar maupun material yang lain. Pada *cutting plan* ini tercantum pula gambar perencanaan posisi penggunaan komponen-komponen pelat yang akan dipotong.

- *Working Drawing*

Adalah gambar bentuk dari blok-blok yang sebenarnya. Untuk bengkel *Fabrikasi* hanya digunakan untuk mengecek kemungkinan terjadi adanya penyimpangan-penyimpangan setelah komponen selesai dan telah dikirim ke bengkel *Assembly*.

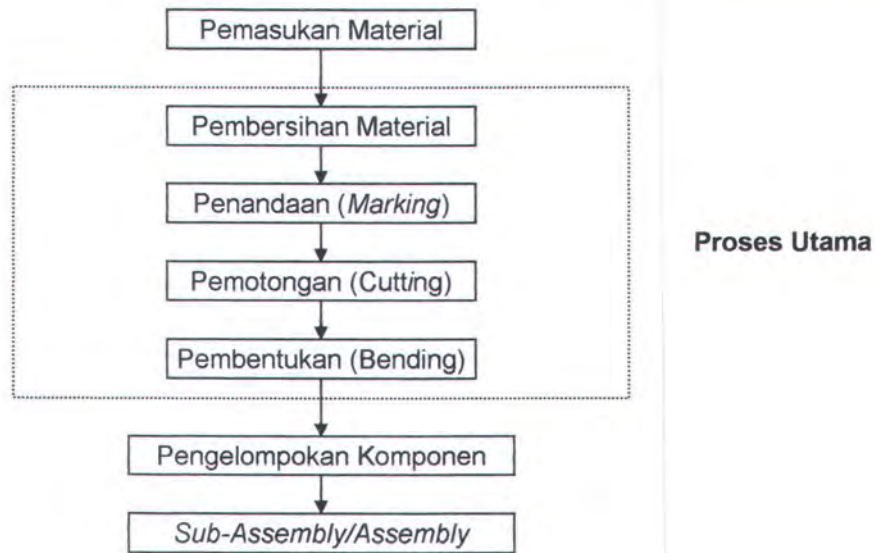
- *Lift dan Scaffold Plece*

Digunakan untuk membuat cincin-cincin (kupingan-kupingan) yang digunakan untuk mengangkat blok-blok yang sudah selesai diassembly untuk diturunkan ke dok. Adapun bentuk dan ukurannya tergantung pada berat blok yang akan diangkat.

Selain dari gambar-gambar di atas bengkel ini juga menerima rambu-rambu yang terdiri dari film dan rambu kayu yang akan digunakan untuk *marking* komponen-komponen yang tidak bisa dijelaskan ukuran dan bentuknya, serta rambu-rambu kayu khusus untuk *bending* dan *fairing*.

Proses yang dilakukan pada bengkel *Fabrikasi* dapat digambarkan seperti pada *Gambar II-3* berikut.



**Gambar II-3**

Urutan proses produksi pada bengkel fabrikasi

Uraian dari proses-proses di atas adalah sebagai berikut :

#### ❑ Proses Pembersihan

Sebelum proses pembersihan ini dilakukan maka terlebih dahulu diadakan identifikasi material yang merupakan suatu tindakan pemeriksaan material yang akan dipakai meliputi:

- *Charge number material*
- Klasifikasi material
- Dimensi material
- Kondisi permukaan material

Setelah itu hasil pemeriksaan material dicatat dalam suatu laporan pemeriksaan (*Cheek Sheet*) sekaligus perbaikannya apabila material tersebut mengalami cacat. Pada pelat yang bergelombang harus diperbaiki dengan pelurusan, dengan menggunakan *roll machine*. Pelat tersebut dilewatkan pada suatu susunan silinder baja. Setelah itu baru dimasukkan dalam *short blasting* dan *primary painting machine*. Alat ini berguna untuk menghilangkan karat dan

kotoran yang melekat pada material dengan menggunakan butiran-butiran baja yang berdiameter  $\pm 0,5-0,8$  mm yang disemprotkan dengan tekanan tinggi, kemudian dicat dengan menggunakan cat dasar.

#### ❑ **Proses Marking**

Proses *Marking* yaitu proses penandaan pada permukaan material yang akan mengalami pekerjaan sesuai dengan ketentuan tanda kerjanya. Pemindahan data *Mould Loft* serta pengukuran-pengukurannya harus dilakukan seakurat mungkin, karena kesalahan pada proses *marking* ini tidak hanya menyebabkan material yang dipersiapkan untuk *assembly* dan komponen-komponen rusak tetapi juga akan menaikkan jumlah material yang terbuang dan pada akhirnya akan mengurangi nilai produktifitas galangan. Langkah-langkah pekerjaan *marking* adalah :

- Pelat diletakkan diatas dantai yang rata, dan dicek apakah material tersebut sesuai dengan yang ada didaftar.
- Rencana pemotongan (*cutting plan*); pada gambar rencana pemotongan ini diusahakan sisa material sekecil mungkin.
- Pembersihan material sebelum di *marking* agar kapur dapat melekat betul.
- Persiapan alat-alat kerja.
- Penandaan atau pemberian nama pada setiap bagian dari material secara jelas agar nantinya tidak tertukar ataupun keliru pada saat perakitan. Nama tersebut disesuaikan dengan kode yang tercantum pada *material list* atau *marking list*. Nama material terdiri dari : nomor kapal, nama kapal, nama komponen dan posisinya (*port side* atau *starboard side*), posisi marking (*up* atau *low marking*)



### ❑ Proses *Cutting* (Pemotongan)

Adalah proses pemotongan material-material yang telah *dimarking* dengan cacatan *marking* tersebut telah disetujui oleh *Quality Assurance*.

Alat-alat pemotongan :

- *Manual Gas Cutting*
- *Semi Automatic Gas Cutting (Scattor)*
- *NC Gas Cutting Machine*
- *Flame Planner Cutting Machine*

### ❑ Proses *Forming* (Pembentukan)

Pada proses *Forming* ini dibedakan antara pelat dan profil. Pada pembentukan material pelat ada dua cara yang dilakukan yaitu :

#### 1. *Heat Forming (fairing)*

Cara ini lebih banyak digunakan untuk bentuk-bentuk tiga dimensi atau sebagai penyempurnaan bentuk dari pelat yang telah dibending dengan mesin. Pada prinsipnya cara ini adalah memanaskan pelat kemudian mendinginkan secara mendadak.

#### 2. *Cold Forming*

Cara ini dengan menggunakan *bending machine*, biasanya untuk bentuk-bentuk yang sederhana. Pada *Cold Forming bending machine* yang digunakan yaitu *roll bending machine* dan *horisontal bending machine*.

Sedang untuk profil, langkah-langkah pembentukan adalah :

- Persiapan material antara lain : profil yang akan *dibending/difairing*, rambu film, landasan untuk *fairing*, *jig* dan lain-lain.
- Gambar ditempatkan dengan kedudukan yang terbalik.



- Dilakukan penitikan pada tempat-tempat tertentu sesuai dengan rambu film. Titik-titik tersebut dihubungkan dengan menggunakan *stroklat* kayu dan *sumitsasi* atau *sumitsubo*.
- Dilakukan pem-bending-an sedikit demi sedikit dan dilakukan pengecekan setiap kali pem-bending-an.

### II.3.3. TAHAP SUB ASSEMBLY

Proses *sub Assembly* ini merupakan proses kelanjutan dari bengkel *fabrikasi*. Pekerjaan pada bengkel ini meliputi :

- Penyambungan pelat
- Pemasangan *Stifener*
- Merakit *Floor*
- Pemasangan *Face Plate*
- Merakit *Web Frame*



Dalam proses pekerjaan diperlukan beberapa data yaitu *yard plan*, *working drawing*, *material list*, *cutting plan*, dan *marking list*.

### II.3.4. TAHAP ASSEMBLY

Pada proses *assembly* ini pekerjaan yang dilakukan adalah merakit panel-panel datar, panel lengkap hingga menjadi panel datar seksi lambung, seksi sekat dan sebagainya sehingga menjadi suatu blok.

#### □ Perakitan Panel

Pada perakitan panel ini, plat-plat diletakkan pada *laticce floor* diatur dan dilas ikat. Urutan pengelasan ikat maupun pengelasannya dimulai dari tengah-tengah panel kemudian secara bertahap keluar, hal ini dilakukan untuk

mengurangi deformasi. Untuk panel-panel yang dibuat dari sambungan banyak pelat dilakukan dengan jalan memberikan beban berat untuk menekan pelat yang akan dilas supaya deformasi yang terjadi sekecil mungkin.

Setelah pelat dilas menjadi satu, pelat diletakkan di atas meja *jig* yang telah disediakan. *Jig* itu harus dicek *levelnya* maupun *countur* dari panel, setelah itu dilakukan juga pemeriksaan posisi *reference line*, terutama untuk *centre line*, *waterline*, dan *framelinenya*. Selanjutnya profil-profil dipasang pada garis-garis *marking* yang telah ditentukan. Profil ini kemudian dilas pada panel.

#### ❑ Perakitan Blok

Perakitan blok ini merupakan kelanjutan dari perakitan panel-panel. Di sini perlu diperhatikan pengaturan letak atau posisi dari seksi-seksi sehingga dapat mengurangi pengelasan yang sulit. Pada umumnya bagian yang datar diletakkan pada bagian bawah.

### II.3.5. TAHAP ERECTION

Pada proses *erection* adalah kelanjutan dari proses-proses sebelumnya yaitu proses *sub assembly* dan *assembly*. Jenis pekerjaan yang dilakukan pada proses *erection* adalah :

#### ❑ Loading

Cara kerja :

- Blok yang ada di pelataran kerja diangkat dengan *crane* yang disesuaikan kapasitasnya.
- Blok yang diangkat harus seimbang dengan meletakkan *ballast* berupa batu cor dengan berat tertentu dan tempat tertentu pula.



- Letak dan besar kupingan harus diperhitungkan.
- Blok ditempatkan pada *keel blok* dan *side blok* yang telah diatur sesuai *marking dock*.

#### □ **Adjusting**

Cara kerja :

- Blok yang baru harus ditempatkan atau diluruskan *center line* dengan blok yang lama.
- Blok yang baru ditempelkan dikerjakan sesuai dengan ukuran yang telah ditentukan setelah *fitting* dan *welding*.
- Mengatur paju pada *keel BLOK* bila kurang tepat atau rapat menempel pada badan kapal atau BLOK agar tidak terjadi gerakan.

#### □ **Fitting**

Cara kerja:

- Bagian *center line* harus lurus antara blok yang satu dengan blok yang lain.
- Melihat bagian sisi-sisi kapal apakah sudah tepat dan sesuai dengan ukuran. Bila tidak, maka dilakukan pengukuran jarak dari *center line* ke sisi badan kapal sesuai dengan gambar kerja. Bila terlalu lebar, maka *frame* dibongkar dan *fairing* atau *bending* menurut gambar kerja. Bila terlalu sempit atau masuk ke dalam juga dilakukan pekerjaan seperti diatas.
- Bila sudah tepat maka ujung-ujung pelat dibentuk kampuh yang sesuai tebal pelat atau gambar kerja.
- Kemudian bila sudah dibuat kampuh las diperiksa lagi kelurusan *center line*, bagian pelat dasar, pelat sisi dan pelat geladak.



- Lalu dilakukan las ikat di tempat-tempat tertentu menurut gambar kerja.

#### □ **Welding**

Cara kerja :

- Setelah di *fitting* dilakukan pemeriksaan oleh QA dan *Class*
- Bila test QA atau *Class* disetujui maka dilakukan pengelasan (*welding*) untuk menyambung blok-blok tersebut tentunya dengan metode dan urutan pengelasan yang sesuai dan benar.
- Selain mengelas pelat sisi, pelat dasar, dan pelat geladak juga dilakukan pengelasan atau penyambungan profil-profil pembujur yang dimulai dari *center line* lalu menepi.

#### □ **Fairing**

Cara kerja :

- Dinding yang cembung dipanasi dengan brander las sampai dengan  $\pm 600^{\circ}\text{C}$ .
- Bagian dinding yang cekung disemprot dengan air untuk menarik struktur material akibat pengerutan material tersebut.
- Pemanasan dilakukan merata berupa titik-titik diseluruh dinding yang mengalami deformasi tersebut diatas.

## II.4. PERMASALAHAN DALAM PROSES PRODUKSI

Selama proses produksi pada pembangunan kapal sedang berjalan terdapat beberapa permasalahan yang timbul pada tiap tahapan proses produksi utamanya dengan terjadinya kesalahan-kesalahan hasil proses produksi, yaitu :

## 1. Proses *Fabrikasi*

### □ *Marking*

Permasalahan pada umumnya disebabkan kesalahan pemberian tanda atau nama, yang terdiri dari :

- tanda pengerjaan
- tanda urutan dan arah pengerjaan
- tanda lokasi komponen
- tanda jenis dan macam komponen
- tanda ukuran dan dimensi komponen

### □ *Cutting*

Macam kesalahan yang terjadi :

- kekasaran permukaan potongan
- penyusutan material karena panas saat pemotongan

### □ *Bending*

Macam kesalahan yang terjadi :

- kesalahan pencocokan rambu bending
- kesalahan sudut bending
- pelat berubah bentuk dengan sendirinya setelah selesai dibending, baik dengan proses dingin maupun panas

## 2. Proses *Sub – Assembly dan Assembly*

### □ *Fitting*

Ketidak telitian akurasi dimensi struktur atau komponen pada pekerjaan *fitting*, umumnya dari kejadian-kejadian berikut :

- *Misalignment* atau ketidaklurusan bagian terpasang.

- Gap atau celah yaitu jarak antara dua bagian yang akan disambung.
- *Misfitting*, yaitu kesalahan tempat pemasangan elemen-elemen pada detailnya.
- Penyimpangan sudut pemasangan antara profil dengan pelat maupun dengan profilnya sendiri.

#### □ *Welding*

Pada proses *welding* akibat perlakuan panas pengelasan pada material umumnya ialah :

- Penyusutan memanjang
- Penyusutan melintang
- *Angular distortion*, pengaruhnya pada penyimpangan sudut pada *fillet weld*.

#### □ *Marking* akhir

Kesalahan penanganan meliputi tanda-tanda :

- posisi fitting
- nama bagian
- letak pada konstruksi hull
- sudut-sudut *fitting*
- proses pekerjaan akhir

### 3. Proses *Erection*

#### □ Penyimpangan Bentuk dan Ukuran

Terdiri dari :

- Penyimpangan bentuk dan ukuran dari perencanaan



Disebabkan oleh deformasi dan kesalahan pemasangan elemen-elemen penyusun konstruksi. Hal ini terjadi karena kesalahan proses pengerjaan maupun material yang digunakan yaitu material pelat dan material las. Macam penyimpangan yang terjadi antara lain :

- Terjadinya ketidak lurusan pada pertemuan sambungan antara blok-blok atau seksi-seksi. Penyebabnya adalah karena adanya penyusutan pada blok atau seksi pada saat proses pembuatannya.
- Terjadinya gap-gap pada sambungan tumpul diantara blok-blok atau seksi-seksi. Penyebabnya adalah karena adanya penyusutan pada waktu proses pembuatan blok atau seksi.
- Terjadinya gap pada sambungan T antara hubungan sekat melintang dengan pelat geladak. Penyebabnya adalah karena adanya penyusutan pada saat pembuatan sekat melintang.
- Terjadinya ketidak lurusan (misalignment) pada konstruksi sekat memanjang. Penyebabnya adalah karena adanya kesalahan pemasangan, deformasi dan perubahan sudut pada sambungan pengelasannya.
- Terjadinya perubahan sudut antara konstruksi lambung dan geladak. Penyebabnya adalah karena adanya perubahan sudut, penyusutan dan deformasi memanjang.
- Terjadinya perubahan sudut antara pelat alas dalam dan pelat lunas. Penyebabnya adalah karena adanya deformasi sudut atau kesalahan pemasangan.
- Terjadinya penyimpangan pelat diantara titik tumpuannya. Penyebabnya adalah karena adanya deformasi sudut.

- Terjadinya penyimpangan sudut antara pelat alas dalam dan sekat melintang. Penyebabnya adalah karena adanya kesalahan pemasangan, deformasi memanjang, deformasi sudut.
- Terjadinya *angular misalignment* pada sambungan tumpul, yaitu pada pelat kulit, pelat alas dalam, pelat sekat dan lain-lain. Penyebabnya adalah karena adanya deformasi sudut.
- Terjadinya penyimpangan dari garis lurus pada pelat kulit, pelat geladak dan lain-lain. Penyebabnya adalah karena adanya deformasi sudut.
- Terjadinya ketidak tepatan pada sambungan pengelasan (*seamslag*) pada pelat alas dalam, pelat sekat dan lain-lain. Penyebabnya adalah karena kurang telitian pemasangan, perencanaan dan deformasi.
- Terjadinya penyimpangan kearah kelebaran dari beban kapal. Penyebabnya adalah karena adanya penyusutan, deformasi sudut, deformasi memanjang dan kesalahan pemasangan.
- Terjadinya distorsi selatif pada blok lambung. Penyebabnya adalah karena adanya deformasi sudut.
- Terjadinya defleksi pada blok lambung. Penyebabnya adalah karena adanya deformasi memanjang.
- Terjadinya lekuk pada pelat kulit. Penyebabnya adalah karena adanya kesalahan atau kurang telitian ukuran, penyusutan.
- Terjadinya lekuk-lekuk pada centre girder, side girder dan penumpu-penumpu. Penyebabnya adalah karena adanya deformasi buckling.



- Terjadinya penggelombangan pada pelat sekat. Penyebabnya adalah karena adanya deformasi buckling, deformasi memanjang.
- Terjadinya penggelombangan pada pelat kulit. Penyebabnya adalah karena adanya deformasi buckling dan perubahan sudut pada penegar-penegarnya.

- Penyimpangan bentuk dan ukuran karena deformasi sudut

Disebabkan perubahan sudut pada sambungan las yang terjadi akibat kesalahan proses pengerjaan dan material yang digunakan.

- Penyimpangan bentuk dan ukuran karena deformasi memanjang

Disebabkan oleh kesalahan pelaksanaan proses pengerjaan material.

- Penyimpangan bentuk dan ukuran karena deformasi *buckling*

Disebabkan oleh gaya dalam (akibat pengelasan dan proses pemotongan *thermal*) maupun gaya luar (adanya beban luar berlebihan).

- Penyimpangan bentuk dan ukuran karena kesalahan pemasangan

Disebabkan kekurangtelitian pada saat pemasangan bagian-bagian konstruksi sebelum dilakukan pengelasan, sehingga konstruksi yang dihasilkan telah berubah dari yang direncanakan.

#### □ *Cocking*

Pada tahap *erection* proses pembangunan kapal ada kecenderungan terjadinya pengangkatan bagian ujung-ujungnya. Pengangkatan bagian *stern* maupun *bow* disebut *cocking up*.

#### □ Cacat

Pengelasan pada tahap *erection* hampir seluruhnya dilakukan secara manual, sehingga tidak mungkin terhindar secara mutlak dari adanya cacat-cacat.



Selain itu juga pelaksanaan pengelasan sangat sulit karena terbentur pada masalah posisi pengelasan yang tidak dapat diubah. Cacat yang terjadi antara lain :

- Cacat yang berasal dari material itu sendiri sebelum digunakan, seperti :
  - *Pitting* ; cacat karena korosi yang terjadi saat penyimpanan.
  - *Scores* (lekuk-lekuk atau takik-takik) ; cacat karena saat transportasi dan penumpukan dalam penyimpanan.
  - *Laminasi* ; cacat karena adanya gelembung-gelembung gas yang terperangkap dan adanya pengerutan pada saat pembuatan.
- Cacat yang berasal dari proses pengelasan
  - *Crack* (retak) ; cacat yang paling berbahaya bila dibandingkan dengan cacat yang lain. Macamnya *crater crack*, *transverse crack* pada *base metal*, *transverse crack* pada *weld metal*, *longitudinal crack*, *under bead crack*, *toe and root crack*, *fusion line crack*, *hot crack*, *lamellar crack*, dan *lamination*.
  - *Cavity* ; cacat karena adanya gelembung yang terperangkap di dalam logam. Macamnya *porosity* dan *shrinkage voids*.
  - *Solid inclusion* ; macamnya *slag inclusion*, *flux inclusion*, *oxide inclusion*, *tungsten inclusion*, *copper inclusion*, *incomplete fusion* dan *incomplete penetration* (cacat karena proses peleburan logam saat pengelasan yang kurang semestinya).
  - *Imperfect shape* atau *unacceptable contour* ; cacat karena adanya bentuk permukaan (*contour*) yang kurang sempurna. Macamnya *under cut*, *under fill*, *over lap*, *excessive reinforcement*, *excessive penetration*, *insufficient throat* dan sebagainya.

- *Miscalleneous defect* ; cacat selain kategori cacat-cacat di atas.
- Cacat yang berasal dari pemotongan dengan gas  
Disebabkan proses pemotongan yang tidak sempurna, sehingga akan menimbulkan atau menghasilkan permukaan potongan yang jelek. Atau karena adanya distorsi pada pelat yang dipotong.
- Cacat yang berasal dari pengerjaan mekanis  
Disebabkan oleh perubahan struktur kristal akibat pengerjaan, baik dingin maupun panas.
- Cacat yang berasal dari kesalahan perencanaan  
Disebabkan oleh kekurangtelitiannya perencanaan ataupun dalam pembentukan detail-detail konstruksi.

Untuk mengurangi kemungkinan terjadinya permasalahan tersebut diatas diperlukan suatu proses pemeriksaan selama proses pembangunan kapal berlangsung. Penjelasan mengenai proses pemeriksaan akan dijelaskan pada bab berikutnya.



**BAB III**  
**PROSES PEMERIKSAAN**  
**BENTUK DAN DIMENSI**



## BAB III

### PROSES PEMERIKSAAAN BENTUK DAN DIMENSI

#### III.1. PEMERIKSAAN PADA TAHAP FABRIKASI

Dalam tahap fabrikasi ini pemeriksaan dilaksanakan pada tiap proses dalam bengkel fabrikasi, yaitu :

##### □ Pemeriksaan Proses *Marking*

Pelaksanaan pemeriksaan dilakukan pada *marking list* berikut gambar-gambarnya, prosesnya, serta hasil *marking*. Pemeriksaan tahap awal dilakukan oleh intern bengkel yaitu *marker* yang disertai oleh kepala regunya dengan prosedur pemeriksaan seperti dalam proses-proses *marking* yang sudah ada. Kemudian kepala bengkel dan kasie melakukan observasi pada hasil-hasil *marking* yang telah diperiksa dan diadakan perbaikan kesalahannya dan memberikan komentar dari hasil obsevasinya dalam *inspection record*.

Untuk *material-marked* yang akan mengalami proses *cutting* harus mendapatkan rekomendasi '*Ok-cut*' dari QA/QC dan *surveyor class*. Untuk hal ini QA melakukan pemeriksaan pendahuluan terhadap *marking-list* dan gambarnya. Kemudian dilakukan pemeriksaan pada *material-marked* dengan pelaksanaan pemeriksaan sesuai ketentuan gambar *marking* dari desain dimana sarana pemeriksaan dengan menggunakan *check-sheet*. Didalam *check-sheet* untuk *accuracy control* bagi tahap fabrikasi tersebut (*marking* dan *cutting*) telah dicantumkan ukuran dimensi sebagai acuan pemeriksaan untuk tiap-tiap bagian atau komponen. Di dalam *check-sheet* ini juga disebutkan siapa yang memeriksa,

bagian yang diperiksa, serta keterangan mengenai ketentuan pemeriksaan seperti sampel yang harus diambil setiap hari. Kemudian juga disediakan kolom keterangan pemeriksa, hal ini berkaitan dengan informasi yang diperoleh dari pihak bengkel. Kemudian pihak QA bersama *surveyor class-owner*, setelah yakin dengan hasil *marking* menyetujui dan mengijinkan dilakukan proses *cutting* dengan memberikan tanda ‘*Ok-cut*’.

Dari uraian di atas urutan pemeriksaan dengan pemeriksa sistem tiga tahap, yaitu :

	Petugas	Warna Kapur
Pemeriksa no. 1	Kasi, Kabeng	Putih
Pemeriksa no. 2	Inspektur Dalmut (divisi QC)	Kuning
Pemeriksa no. 3	Surveyor QC/A	Hijau
Final Inspeksi	Surveyor <i>class</i> dan <i>surveyor owner</i>	Merah

**Tabel III-1**  
Urutan pemeriksaan tiga tahap

*Surveyor QC/A* dapat menggunakan kapur merah apabila pemeriksaan dilakukan bersama-sama dengan *surveyor class* atau *surveyor owner*.

□ **Pemeriksaan Proses *Cutting***

Pemeriksaan pertam-tama dilakukan oleh pekerja itu sendiri dimana bila terdapat perbaikan-perbaikan yang perlu dilaksanakan harus disetujui oleh kabeng. Hal ini berkaitan dengan schedule produksi dan ketentuan prosedur pekerjaan sebagai standarisasi pekerjaan perbaikan serta ketentuan akan besar *exess material* dan margin. Untuk keperluan pemeriksaan atau evaluasi oleh QC/A, kabeng membuat dan menyusun catatan pemeriksaan hasil pemeriksaan



yang berisikan hasil pemeriksaan akurasi dimensi, prosedur-prosedur perbaikan, serta kondisi kondisi pekerjaan dan peralatan yang digunakan.

Untuk urutan pemeriksaan pada proses *cutting* seperti yang diuraikan sebelumnya dalam proses *marking* yaitu sesuai dengan urutan proses *cutting* yang benar.

#### ❑ Pemeriksaan Proses Bending

Seperti halnya *marking* dan *cutting*, prosedur pemeriksaan proses bending dilakukan dalam tiga tahap yaitu ; pekerjaan kepala regu, kabeng-kasi dan divisi QC (Dalmut), hal ini untuk intern bengkel , kemudian dilanjutkan oleh *surveyor* QC/A untuk sistem informasi standar dan persiapan *surveyor class-owner*. Semua informasi mengenai metode dan jadwal pemeriksaan diberikan oleh bagian QA/C dimana semuanya disusun dalam suatu *check sheet*. Dan urutan pemeriksaannya sama dengan urutan pemeriksaan pada proses-proses sebelumnya.

Dari penjelasan di atas dapat dikatakan bahwa untuk semua proses pada tahap fabrikasi prosedur pemeriksaan, dalam hal ini urutan pemeriksaan dan siapa yang melaksanakan, adalah sama, yaitu :

- Awal.....pekerja ; perbaikan dengan rekomendasi kabeng  
(*record* kondisi dan penyimpangan).
- Periksa I.....Kasi dan Kabeng ; kapur putih + *check sheet* (*record* data  
penyimpangan).
- Periksa II.....Divisi QC ; kapur kuning + *check sheet* (*record* data  
penyimpangan dan keterangan).



- Periksa III.....Surveyor QC/A ; kapur hijau + *check sheet* (record data penyimpangan dan keterangan).
- Periksa Tahap Akhir, *surveyor class* dan *owner* ; kapur merah + *check sheet* (record data penyimpangan dan keterangan).

### III.2. PEMERIKSAAN PADA TAHAP *SUB ASSEMBLY*

Seperti telah disebutkan sebelumnya bahwa pada pekerjaan *sub assembly* terdapat kegiatan-kegiatan ; *fitting*, *welding*, dan *marking akhir-finishing*. Maka pelaksanaan pemeriksaan bentuk dan dimensi komponen dilakukan pada tiap kegiatan tersebut, dimana fokus perhatian pada pelaksanaan proses *fitting*. Hal ini dikarenakan kegiatan *fitting* merupakan pekerjaan penentu dari posisi lanjut dari *members/parts* yang dipasang, dimana bila terjadi kesalahan memerlukan waktu dan biaya operasi yang cukup besar.

Beberapa petunjuk yang dapat dipakai sebagai pertimbangan pelaksanaan pemeriksaan pada tahap *sub assembly* adalah sebagai berikut :

- Pekerjaan pada tahap ini merupakan langkah awal dalam memproses potongan pelat-pelat datar yang sesuai dengan gambar desain menjadi bentuk-bentuk tiga dimensi (3D). Jika presisi dari dimensinya tidak terjamin, maka akan timbul berbagai masalah pada proses pekerjaan selanjutnya.
- Tidak cukup hanya memeriksa gambar-gambar, tetapi lebih penting untuk dapat mengartikan secara tepat dan kemudian memilih prosedur dan metode *assembly* yang efisien.
- Memeriksa tabel *members* atau *parts* untuk menentukan jumlah dan jenis *members*.

### III.3. PEMERIKSAAN PADA TAHAP *ASSEMBLY*

Pemeriksaan yang dilakukan pada tahap *assembly* dilaksanakan pada proses pekerjaan sebagai berikut :

#### □ Pemasangan dan penyambungan pelat

- Pemeriksaan terhadap material pelat :
  - Pemeriksaan simbol-simbol pada pelat ; nama, nomor, dan lokasi pelat.
  - Pemeriksaan terhadap marking dan posisi pelat ; P/S.
  - Pemeriksaan terhadap sudut *bavel*.
- Pemeriksaan bentuk alur pengelasan *butt joint* terutama terhadap sudut-sudut *bavel* kampuh las dengan kesesuaiannya dengan besar gap dan keseragaman lebarnya sepanjang alur las tersebut.
- Pemeriksaan kelurusan atau kedataran permukaan pada sisi sambungan dengan menggunakan *face alignment piece* disertai pengukurannya dengan *water pass*.
- Komponen yang dihasilkan dari proses pengelasan tersebut diperiksa lagi yang meliputi kelurusan atau kedataran permukaan pelat dan bentuk puntiran sambungan. Pemeriksaan dengan menggunakan pengukuran lebar, panjang, dan diagonal dari proyeksi bentuk komponen tersebut disertai dengan *levelling* dengan *water pass*.

#### □ Marking dan Cutting

- Pengukuran lebar sambungan pelat yang disesuaikan dengan ketentuan pada gambar kerja.



- Penandaan *basic frame line* pada arah memanjang dengan memeriksa ukuran diagonal sambungan pelat dan menandai *basic frame line* pada titik temu secara *orthogonal* ke arah memanjang di tengah-tengah blok.
- Pengukuran panjang dari *basic frame line*, sesuai ukurannya.
- Pemeriksaan *marking* dan *cutting* seperti pada tahap fabrikasi.

#### □ **Welding**

- Pengukuran ketegaklurusan *fitting* pembujur dan pelintang terhadap pelat yang dipasang.
- Pemeriksaan hasil pengelasan terhadap akibat deformasi pengelasan yaitu distorsi ataupun puntiran.
- Pemeriksaan puntiran dengan cara memberi tanda pada ketinggian tertentu di setiap ujung-ujung pembujur dengan menggunakan *water level*.
- Pemeriksaan terhadap cacat-cacat akibat pengelasan, dimana pengaruhnya terhadap akurasi bentuk dan dimensi terletak pada penyusutan ukuran karena proses perbaikannya.

### III.4. PEMERIKSAAN PADA TAHAP *ERECTION*

Proses pemeriksaan bentuk dan dimensi pada tahap *erection* ini diutamakan pada daerah sambungan antar *block* maupun *section*. Pelaksanaan pemeriksaan secara global dilakukan pada hasil akhir pengelasan *erection*, yaitu terhadap ukuran-ukuran utama kapal setelah penyambungan semua blok dan seksi telah selesai.



Prosedur pelaksanaan pemeriksaan tetap sama dengan tahap sebelumnya yaitu dengan pemeriksaan tiga tahap, kabeng/kasie, divisi QC, *surveyor QC/A* ditambah dengan pemeriksaan *surveyor class/owner*. Sedangkan sarana yang dipakai selama proses pemeriksaan adalah tetap *check sheet* dan sebagai pedoman akurasi menggunakan standar yang telah ditetapkan.

Proses pemeriksaan komponen kapal selama proses pembangunan kapal seperti diuraikan diatas adalah yang digunakan selama ini. Untuk itu penulis mencoba mengembangkan dengan suatu cara dengan menggunakan komputer grafis. Hal ini digunakan memvisualisasikan komponen 3-D kapal dan dapat dipakai untuk proses analisa lebih lanjut. Konsep dasar mengenai komputer grafis akan dijelaskan pada bab berikutnya.

**BAB IV**  
**KONSEP DASAR KOMPUTER**  
**GRAFIS**



## **BAB IV**

### **KONSEP DASAR KOMPUTER GRAFIS**

#### **IV.1. KONSEP DASAR BENDA 3D**

Jika kita menampilkan suatu bentuk 3-D, lebih banyak pertimbangan, dibandingkan dengan bentuk 2-D, yang harus diperhitungkan selain memasukkan nilai koordinat 3-D. Batasan-batasan obyek dapat dibentuk oleh berbagai kombinasi bentuk permukaan datar (*plane*) dan kurva (*curve*), dan kadangkala dibutuhkan informasi khusus tentang obyek interior. Menampilkan transformasi dalam bentuk 3-D adalah hal yang sangat kompleks karena kita membutuhkan lebih banyak parameter yang dipilih untuk memberikan spesifikasi bagaimana bentuk 3-D bisa terwakili pada sebuah media tampilan (*display device*). Penjabaran bentuk komponen harus diproses melalui tampilan transformasi koordinat dan proyeksi berulang untuk merubah tampilan koordinat 3-D kedalam media koordinat 2-D. Bagian benda yang terlihat untuk pandangan yang dipilih, harus diidentifikasi, dan algoritma *rendering* permukaan harus digunakan, jika ingin menampilkan bentuk yang nyata.

#### **IV.2. PENYAJIAN OBYEK 3-D**

##### **IV.2.1. PERMUKAAN POLIGON (*POLYGON SURFACES*)**

Kebanyakan keterbatasan penyajian obyek 3-D terletak pada kesatuan poligon-poligon permukaan yang membatasi interior obyek. Banyak sistem grafik



memberikan semua penjabaran obyek sebagai kesatuan-kesatuan poligon-poligon permukaan. Hal ini mempermudah dan mempercepat *rendering* permukaan dan penampilan obyek, tetapi terbatas pada permukaan-permukaan dengan persamaan linier. Dengan alasan ini penjabaran obyek kebanyakan didasarkan pada “grafik standar obyek”.

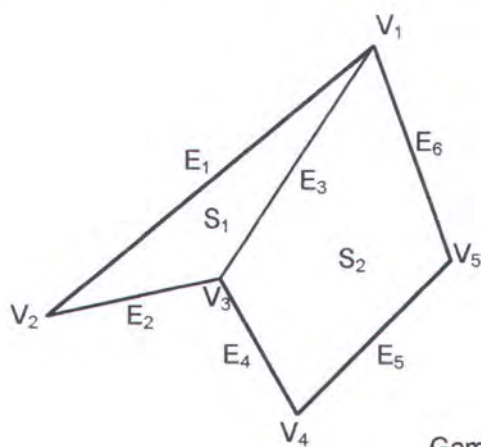
Penyajian suatu poligon dalam penggambaran *polyhedron* dapat secara tepat mendefinisikan permukaan bentuk muka. Tetapi untuk menggambarkan bentuk-bentuk tertentu, contohnya silinder, masih terbentuk sebagai jaring poligon. *Rendering* digunakan dengan menginterpolasi bentuk-bentuk bayangan permukaan poligon untuk mengurangi prosentase batasan ujung poligon.

Kita dapat menentukan suatu permukaan poligon dengan satu set koordinat ujung sudut (*vertex*) dan parameter-parameter yang berhubungan. Sebagai informasi poligon adalah input, data ditempatkan di dalam tabel-tabel yang digunakan untuk proses berikutnya, tampilan, dan memanipulasi suatu obyek yang disebut dengan “tabel poligon”. Tabel data poligon dapat dikelompokkan menjadi dua, yaitu tabel geometri dan tabel atribut. Tabel data geometri berisi koordinat ujung sudut dan parameter-parameter untuk mengidentifikasi sifat ruang dari permukaan poligon. Informasi atribut sebuah obyek terdiri dari parameter-parameter yang digunakan untuk menentukan tingkat kejelasan atau ketajaman obyek, bayangan permukaan, dan karakteristik susunan bentuk. Data geometri diberikan dalam tiga bagian (*Tabel IV-1*) yaitu :

1. Tabel ujung sudut (*vertex*), berisi nilai koordinat.
2. Tabel ujung (*edge*), berisi tambahan petunjuk bagi tabel vertek untuk mengidentifikasi bentuk ujung sudut tiap ujung poligon.

3. Tabel poligon (*polygon*), berisi tambahan petunjuk bagi tabel ujung untuk mengidentifikasi ujung-ujung tiap poligon.

Diilustrasikan dengan *Gambar IV-1* berikut :



**Gambar IV-1**  
Gambar dua permukaan poligon yang saling bersinggungan

TABEL VERTEX	TABEL EDGE	TABEL PERMUKAAN POLIGON
V <sub>1</sub> : x <sub>1</sub> , y <sub>1</sub> , z <sub>1</sub> V <sub>2</sub> : x <sub>2</sub> , y <sub>2</sub> , z <sub>2</sub> V <sub>3</sub> : x <sub>3</sub> , y <sub>3</sub> , z <sub>3</sub> V <sub>4</sub> : x <sub>4</sub> , y <sub>4</sub> , z <sub>4</sub> V <sub>5</sub> : x <sub>5</sub> , y <sub>5</sub> , z <sub>5</sub>	E <sub>1</sub> : V <sub>1</sub> , V <sub>2</sub> E <sub>2</sub> : V <sub>2</sub> , V <sub>3</sub> E <sub>3</sub> : V <sub>3</sub> , V <sub>1</sub> E <sub>4</sub> : V <sub>3</sub> , V <sub>4</sub> E <sub>5</sub> : V <sub>4</sub> , V <sub>5</sub> E <sub>6</sub> : V <sub>5</sub> , V <sub>1</sub>	S <sub>1</sub> : E <sub>1</sub> , E <sub>2</sub> , E <sub>3</sub> S <sub>2</sub> : E <sub>3</sub> , E <sub>4</sub> , E <sub>5</sub> , E <sub>6</sub>

**Tabel IV-1**  
Data geometri yang menyajikan dua permukaan poligon yang saling bersinggungan, dibentuk oleh enam *edge* dan lima *vertex*

Informasi geometri tambahan biasanya diberikan dalam tabel data termasuk kemiringan (*slope*) untuk tiap ujung (*edge*) dan koordinat luasan tiap poligon. Kita dapat menghitung kemiringan ujung (*edge*), dan mencari nilai koordinat untuk mengidentifikasikan nilai minimum dan maksimum x, y, dan z untuk tiap-tiap poligon. Kemiringan ujung dan informasi poligon diperlukan dalam proses berikutnya, sebagai contoh, *rendering* permukaan.

Pengecekan kesalahan dalam proses tetap diperlukan agar diperoleh hasil yang sesuai dengan keinginan. Oleh karena itu dibutuhkan informasi yang lengkap mengenai diskripsi obyek untuk memudahkan pengecekan. Hal ini bisa

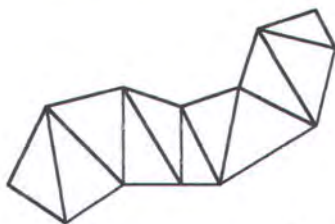


dilaksanakan jika ketiga tabel data (*vertex*, *edge*, dan poligon) digunakan dalam diskripsi obyek. Beberapa ketentuan yang terdapat dalam kemasan grafis (*graphic package*) adalah sebagai berikut :

- Setiap ujung sudut (*vertex*) ditabelkan sebagai titik ujung bagi sedikitnya dua ujung garis.
- Setiap ujung (*edge*) adalah bagian dari sedikitnya satu poligon.
- Setiap poligon saling berdekatan.
- Setiap poligon mempunyai paling tidak satu tepi yang bersinggungan.
- Jika tabel ujung (*edge*) mengandung petunjuk poligon, setiap tepi mengacu terhadap petunjuk poligon yang saling bertemu.

#### IV.2.2. JARING-JARING POLIGON (*MESH*)

Beberapa kemasan grafis memberikan fungsi-fungsi poligon untuk memodelkan obyek. Suatu permukaan bidang dapat ditentukan dengan sebuah fungsi seperti *fillArea*. Tetapi jika permukaan-permukaan obyek disusun, penggambaran obyek lebih mudah digunakan dengan memakai fungsi jaring (*mesh*). Salah satu bentuk jaring poligon adalah jalur segitiga seperti *Gambar IV-2* berikut :



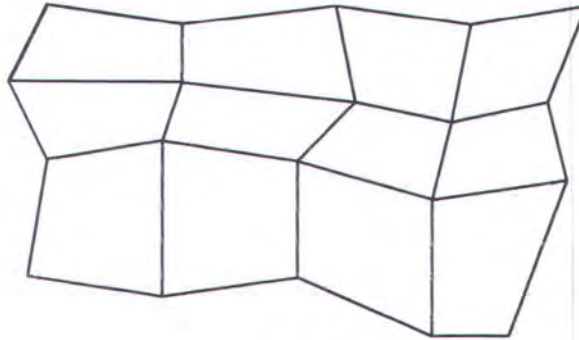
**Gambar IV-2**

Jaring segitiga yang dibentuk oleh 11 segitiga menghubungkan 13 vertex

Fungsi tersebut menggambarkan  $n - 2$  segitiga yang saling berhubungan, menyatakan koordinat-koordinat untuk ujung sudut (*vertex*)  $n$ . Fungsi lain yang sama adalah jaring segiempat, yang menggambarkan sebuah jaring  $(n - 1)$  kali



$(m - 1)$  segiempat, memberikan koordinat untuk sebuah  $n \times m$  barisan ujung sudut (*vertex*), seperti pada *Gambar IV-3* berikut :



**Gambar IV-3**

Jaring empat sisi mengandung 12 segiempat yang dibentuk dari  $5 \times 4$  array *vertex* input

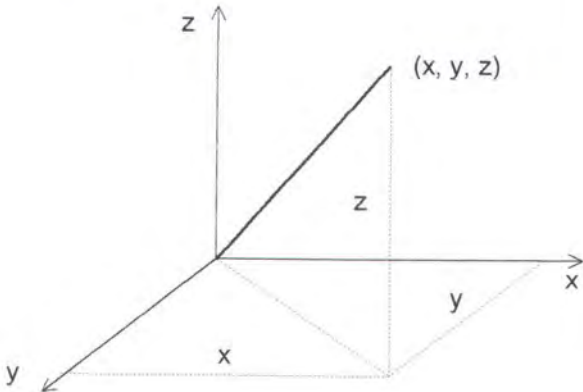
Jika poligon-poligon terdiri lebih dari tiga ujung sudut (*vertex*), dimungkinkan ujung sudut (*vertex*) tersebut tidak semua digunakan dalam satu bidang. Hal ini dapat menyebabkan kesalahan numerik atau kesalahan dalam pemilihan posisi koordinat ujung sudut (*vertex*). Suatu cara yang mudah untuk mengatasi masalah ini yaitu membagi poligon-poligon menjadi bentuk segitiga.

#### IV.2.3. SISTEM KOORDINAT *CARTESIAN* –3D

Dengan dibutuhkannya input data yang berupa titik-titik koordinat 3-D dalam penyajian obyek 3-D dengan komputer grafis, maka dibutuhkan suatu sistem koordinat ruang (3-D).

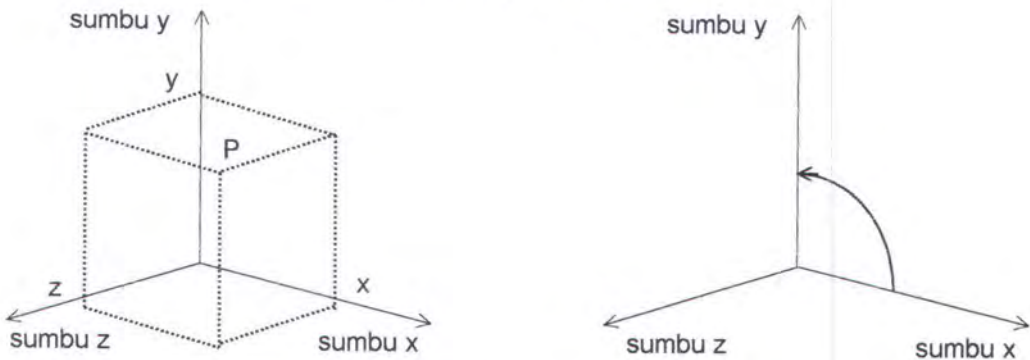
Kemasan grafis (*graphic package*) biasanya membutuhkan apa yang dinamakan paramater-parameter koordinat yang memberikan spesifikasi terhadap sumbu koordinat *Cartesian*. Tetapi pada beberapa aplikasi, sistem koordinat non-*Cartesian* juga digunakan. Bola (*spherichal*), silinder (*cylindrical*), atau bentuk-bentuk simetris lainnya sering dimasukkan dalam penjabaran gambar atau manipulasi obyek. Hal ini hanya dapat dilakukan jika menggunakan sistem grafis

khusus. Hal utama yang harus kita lakukan adalah merubah diskripsi non-*Cartesian* suatu obyek kedalam koordinat *Cartesian*. Disini akan digunakan sistem koordinat *Cartesian* geometri. Untuk sistem koordinat 3-D digunakan tiga komponen titik dengan masing-masing sumbu koordinat x, y, dan z, seperti pada *gambar IV-4* di bawah ini.



**Gambar IV-4**  
Sistem koordinat tiga dimensi

Atau dapat digunakan sistem koordinat tangan-kanan.



**Gambar IV-5**  
Koordinat Cartesian dengan kaidah tangan kanan

*Gambar IV-5* menunjukkan sumbu koordinat konvensional pada sistem *Cartesian* 3-D yang dinamakan sistem tangan-kanan karena titik-titik pada ibu jari tangan-kanan dalam arah sumbu positif z dengan membayangkan arah putaran jari-jari dari sumbu positif x ke sumbu positif y (sepanjang 90°).



### IV.3. KONSEP *COMPUTER AIDED DESIGN (CAD)*

Sebuah gambar teknik hanya merupakan penyajian suatu obyek dalam bentuk gambar tidak ditujukan untuk memberikan gambaran visual secara lengkap dan terperinci sesuai detail bentuk, warna dan konturnya. Pandangan-pandangan dalam gambar itu sendiri merupakan gabungan penyajian bentuk 2-D dari obyek 3-D. Bagaimana pandangan-pandangan ini dituangkan kertas gambar diatur oleh ketentuan dan variasi standar. Dapat dikatakan prosedur penggambaran yang digunakan telah ditentukan dengan satu set peraturan yang harus diikuti untuk membuat gambar, seperti layout ujung pertama, ujung ketiga dan sistem proyeksi pada umumnya.

Sekarang, dengan kebangkitan *computer-aided design*, dimungkinkan untuk mengembangkan aturan-aturan yang ada dengan cara lain dan digunakan sebagai prosedur penggambaran 2-D yang berasal dari model 3-D.

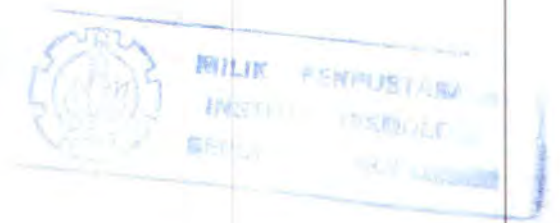
Secara tradisional, seorang tukang gambar kesulitan menggambarkan pandangan dengan jelas suatu obyek (depan, samping, dan atas). Dengan *CAD* model 3-D dapat digambarkan secara jelas dan nyata, berbagai pandangan melalui segala posisi yang diinginkan dengan tampilan terbaik sesuai dengan tujuan yang diperlukan.

*CAD* database mengambil alih aturan dan cara menggambar berbagai bentuk obyek dan tujuan, telah menjadi alat komunikasi yang berisi informasi yang dibutuhkan untuk seorang pengguna (tukang gambar). Suatu data gambar kompleks dapat dibagi menjadi beberapa bagian data yang diutamakan untuk mempermudah penggunaan dalam proses penggambaran. Gambar dapat dirubah bentuk (konfigurasi ulang) dengan tujuan memberikan koordinat geometri untuk setting perintah dan pengoperasian mesin, memberikan keterangan terpisah



mengenai ketentuan dan keterkaitan dalam operasi *assembly* atau memberikan titik acuan (*reference*) koordinat dan koordinat lainnya yang digunakan untuk pengecekan suatu obyek yang telah digambar.

Konsep-konsep tersebut diatas merupakan dasar yang akan digunakan untuk pembuatan prototipe *matching process* yang akan dijelaskan pada bab berikutnya.



**BAB V**  
**PEMBUATAN PROTOTIPE**  
**MATCHING PROCESS**



## BAB V

### PEMBUATAN PROTOTIPE *MATCHING PROCESS*

#### V.1. BENDA KERJA SEBAGAI PROTOTIPE

Pada Tugas Akhir ini, telah dipilih 1 (satu) kapal yang masih dalam tahap pembangunan di PT. (Persero) PAL INDONESIA sebagai salah satu prototipe komponen kapal, yaitu *Kapal Penumpang PAX 500* Tahap II, yang mempunyai ukuran utama kapal adalah sebagai berikut :

<i>Length Over All (LOA)</i>	=	74,00	m
<i>Length Perpendicullar (Lpp)</i>	=	68,00	m
<i>Breath Moulded (Bmld)</i>	=	15,20	m

Untuk beberapa *parts/members* yang kami ambil salah satu dari komponen PAX 500, pengerjaannya menggunakan skala 1:3. Dalam pembuatan komponen kapal ini kami membuat empat bagian yang dapat mewakili bentuk-bentuk komponen kapal secara umum. Pemilihan PAX 500 sebagai studi kasus karena dalam pembangunannya, kapal ini masih terjadi *miss alignment* yang merupakan salah satu kesalahan penyimpangan bentuk dan dimensi.

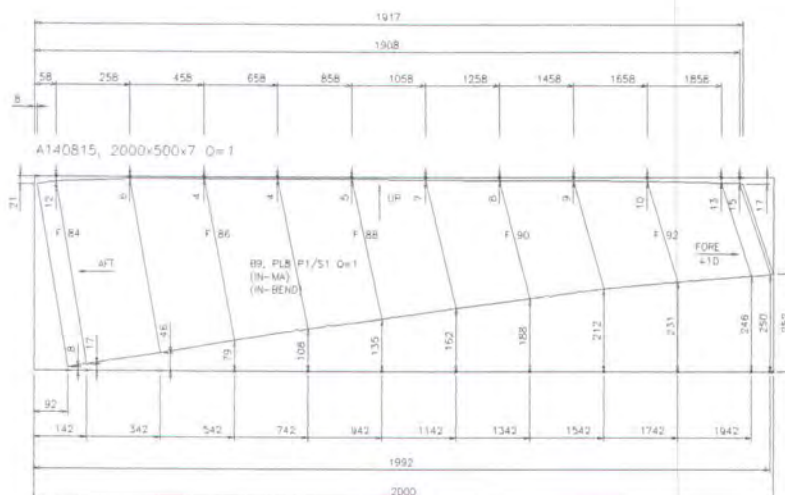
Pembuatan komponen kapal PAX 500 tersebut sebagai kasus, dengan mengacu pada gambar bukaan kulit untuk dapat mencocokkan posisi *parts* yang akan dibuat, melihat kelengkungan *parts* untuk bentuk lengkung pada gambar *body plan*, membuat rambu untuk pengerjaan komponen lengkung, serta menyiapkan rambu beserta *working drawing* untuk proses selanjutnya. Material pelat yang digunakan untuk pembuatan komponen tersebut mempunyai tebal 6

mm. Benda uji tersebut diberi tanda sesuai dengan gambar *design* yang diterima dari *design office*. Pemberian tanda tersebut sebagai garis referensi pengukuran. Contoh gambar kerja komponen lengkung tidak simetris seperti pada Komponen 3-D yang digunakan sebagai kasus dimana dapat mewakili jenis komponen dalam sebuah kapal pada umumnya. Untuk itu penulis menggunakan gambar bukaan kulit, marking list sebagai referensi, dan kami memilih 4 (empat) komponen yaitu :

- pelat datar
- pelat lengkung simetris, No. A140402
- pelat lengkung tidak simetris, No. A140425
- *inner part*, No. A140815

(gambar kerja komponen pada lampiran A)

Komponen yang digunakan sebagai contoh kasus pelaksanaan *matching process* adalah pelat lengkung tidak simetris, (No. A140425). Gambar kerja komponen ini terdapat pada *Gambar V-1* berikut.



**Gambar V-1**  
Penandaan bilga gading no. 84 - 93 PAX 500



Penggambaran komponen-komponen 3-D kapal dalam *matching process* menggunakan komputer grafis. Data *input* yang digunakan berupa koordinat tiga dimensi yang disebut *vertex* dimana bila titik-titik ini digambarkan dapat membentuk suatu gambar grafis komponen 3-D kapal. Dua macam data yang diperlukan untuk pelaksanaan *matching process*, yaitu :

#### □ Data Desain

Data desain berisikan titik-titik koordinat 3-D yang diambil dari gambar-gambar desain pembuatan suatu komponen kapal. Pelaksanaan pembacaan data desain memerlukan beberapa gambar yaitu; gambar kerja, *body plan*, dan bukaan kulit. Adapun cara-cara pembacaan data desain terdapat pada lampiran B.

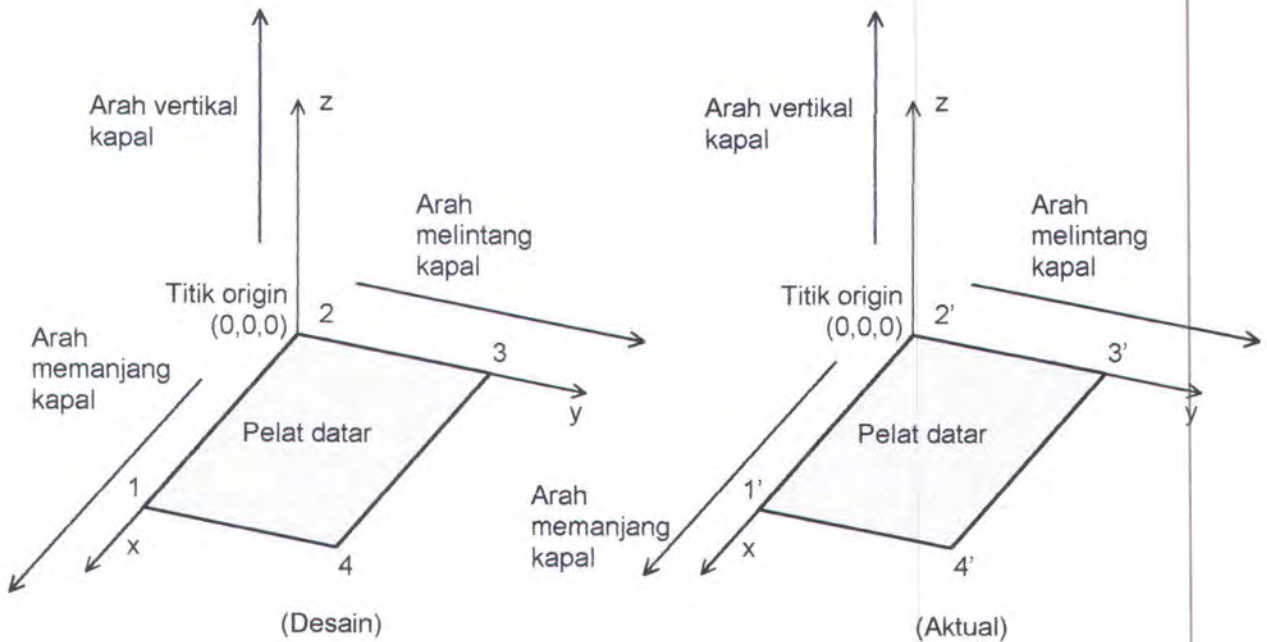
#### □ Data Aktual

Data aktual yang digunakan berdasarkan pengukuran keempat contoh kasus komponen kapal PAX 500. Pengukuran dilaksanakan di Laboratorium Hidrodinamika Indonesia (LHI) menggunakan alat ukur yang dinamakan *marking table* (penjelasan mengenai *marking table* dapat dilihat pada lampiran C). Hasil pengukuran menggunakan alat ini berupa data digital nilai titik-titik koordinat 3-D (x, y, z).

Titik-titik yang diukur dibuat berdasarkan garis-garis bantu dari gambar-gambar kerja berupa *marking line* ataupun *bending line* sebagai pedoman pembuatan dan pembentukan komponen kapal di bengkel fabrikasi yang berpotongan dengan gading. Sehingga titik-titik koordinat baik desain maupun aktual mempunyai posisi yang sama dengan sistem satuan dimensi harus sama.

## V.2. SISTEM KOORDINAT *MATCHING PROCESS*

Ketentuan sistem koordinat pada *matching process* menggunakan sistem koordinat lokal diilustrasikan sebagai berikut :



**Gambar V-2**

Penentuan titik origin koordinat 3-D *Matching Process*

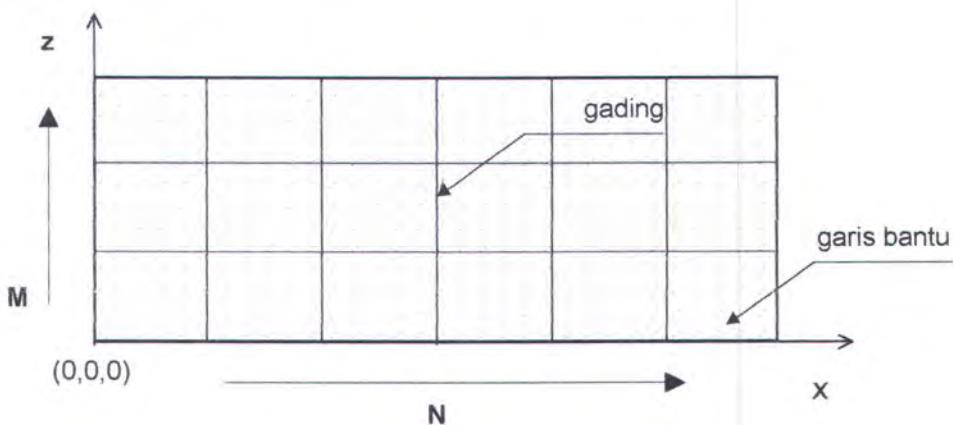
- Titik yang diasumsikan sebagai titik origin sumbu koordinat 3-D antara desain dan aktual kapal harus mempunyai letak yang sama, seperti pada *Gambar V-3*. Koordinat salah satu ujung pelat adalah (0,0,0).
- Titik origin untuk komponen dengan letak melintang kapal pada salah satu ujung pelat sesuai dengan bentuk masing-masing komponen.
- Arah sumbu x untuk desain dan aktual sesuai dengan arah memanjang kapal.
- Arah sumbu y untuk desain dan aktual sesuai dengan arah melintang kapal.



- Arah sumbu z untuk desain dan aktual sesuai dengan arah vertikal kapal.
- Posisi komponen saat pengukuran dan penggambaran sesuai dengan posisi aslinya pada konstruksi kapal.

### V.3. 3D POLYGON MESHES

Dalam *matching process* penggambaran grafis menggunakan software AutoCAD. Metode penggambaran yang digunakan adalah 3-D *polygon mesh* atau dengan sebutan *3D mesh*. Suatu *polygon mesh* terdiri dari suatu matrik  $M \times N$  dimana  $M$  dan  $N$  berfungsi untuk menentukan posisi kolom dan baris yang membentuk *vertex*. *Polygon meshes* dapat berbentuk terbuka atau tertutup. Suatu *mesh* dikatakan terbuka jika garis-garis kolom atau barisnya tidak saling berhubungan. Sebaliknya untuk *polygon meshes* tertutup jika jika garis-garis kolom atau barisnya saling berhubungan. Aturan *meshes*  $M \times N$  ini digunakan untuk urutan memasukkan data *input*.



**Gambar V-3**  
Arah M dan N pada pelat kulit

Arah M dan N tergantung pada posisi komponen pada lambung kapal dan letak titik origin (0,0,0). Titik M x N (0,0) terletak pada titik origin komponen. Untuk komponen memanjang kapal arah M dari titik origin komponen (0,0,0) ke arah memanjang kapal mengikuti bentuk komponen yang sesuai sumbu positifnya. Sedangkan arah N dari titik origin komponen dan mengikuti bentuk komponen (memanjang ataupun vertikal) sesuai sumbu positif. Untuk komponen melintang kapal arah M dari titik origin komponen ke arah melintang kapal mengikuti bentuk komponen yang sesuai sumbu positifnya. Sedangkan arah N dari titik origin komponen dan mengikuti bentuk komponen (biasanya vertikal, komponen yang melintang misalnya sekat atau gading) sesuai sumbu positif. Atau lebih mudahnya bahwa arah M sesuai dengan arah horisontal pelat dan arah N sesuai dengan arah vertikal pelat atau arah tegak lurus M dan M x N dari arah titik origin komponen ke arah sumbu positifnya masing-masing, seperti pada *Gambar V-3* untuk komponen memanjang kapal (pelat lambung).

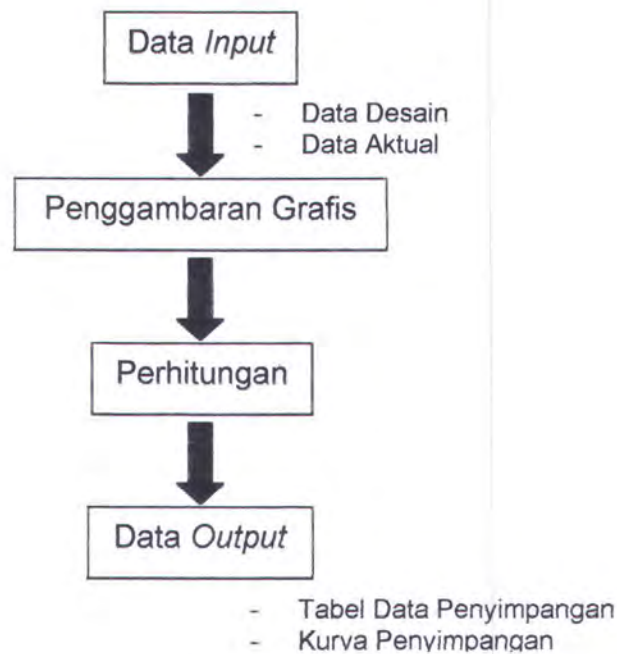
#### V.4. KONSEP LANGKAH *MATCHING PROCESS*

Konsep program *matching process* didasari oleh fenomena gravitasi bumi. Jika suatu benda dijatuhkan dari suatu ketinggian tertentu dari tanah, maka benda tersebut bergerak vertikal ke bawah dengan lintasan yang lurus dengan asumsi tidak ada gaya luar yang mengakibatkan benda bergeser dari lintasannya, sampai ke tanah tanpa ada perubahan bentuk dan ukuran. Berdasarkan hal tersebut dalam *matching process* untuk mengetahui penyimpangan bentuk dan ukuran suatu komponen 3-D dengan cara menggerakkan gambar komponen 3-D aktual ke arah gambar desain yang mempunyai kedudukan tetap (tidak bergerak)



dengan menggunakan visualisasi komputer grafis. Apabila kedua gambar tersebut sudah saling berimpit, mungkin akan terdapat beberapa bagian komponen yang tidak berimpit menjadi satu. Bagian inilah yang dihitung selisihnya, sehingga akan diperoleh data penyimpangan bentuk dan ukuran komponen tersebut.

Perwujudan konsep *matching proses* diatas dijabarkan dengan langkah-langkah yang dapat digambarkan dengan diagram berikut.



**Gambar V-4**  
Langkah *Matching Process*

Uraian dari *Gambar V-4* adalah sebagai berikut :

#### □ **Data Input**

*Input* ini digunakan untuk proses pemasukan dan pembacaan data yang dimasukkan oleh pemakai. Disusun dengan menggunakan bahasa pemrograman *Visual BASIC*. *Data input* yang digunakan berupa koordinat tiga dimensi yang disebut *vertex* tiga dimensi dimana bila titik-titik ini digambarkan dapat membentuk suatu gambar grafis komponen 3-D kapal. Oleh karena itu hal yang perlu diperhatikan adalah cara pemasukan data yang berhubungan dengan metode penggambaran *3D mesh* pada *AutoCAD*. Langkah pertama yang dilakukan yaitu

memasukkan jumlah baris (M) dan jumlah kolom (N) komponen dengan besar antara 2 sampai 256. Sebelumnya komponen dibagi dengan sejumlah garis bantu M dan N yang mempunyai jumlah titik  $M \times N$  (*vertex*). Kemudian langkah kedua yaitu memasukkan titik-titik *vertex* tersebut sesuai urutan yang diberikan oleh program *input*. Data *input* yang diperlukan untuk pelaksanaan *matching process* terdiri dari data desain dan data aktual.

Selain itu masih diperlukan suatu data *input* lagi yang dipakai dalam proses perhitungan dalam *matching process* yaitu toleransi. Toleransi merupakan suatu batasan dimana besar selisih antara data desain dan data aktual dapat dikatakan benar. Toleransi ini diperlukan karena pada kenyataannya kemungkinan selisih tiap titik koordinat sama dengan nol sangat kecil. Apabila selisih yang terdapat pada titik-titik koordinat 3-D tersebut di luar batasan toleransi, maka pada posisi itu terjadi penyimpangan bentuk. Sedangkan apabila selisihnya masih dalam batasan toleransi diambil yang terkecil sebagai dasar perhitungan mencari besar penyimpangan pada semua titik koordinat komponen.

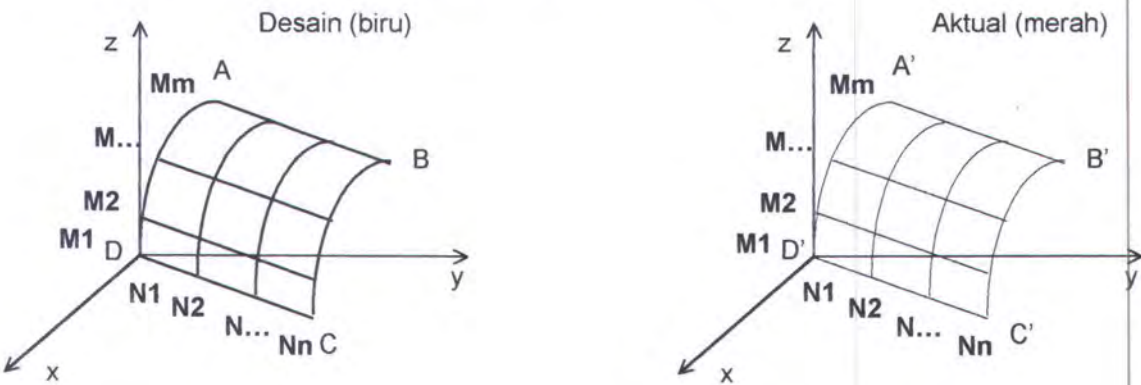
#### □ **Penggambaran Grafis *Matching Process***

Penggambaran komponen dalam *matching process* ini menggunakan utility yang ada pada *AutoCAD* dengan bahasa pemrograman *LISP*. Untuk membedakan kedua gambar, penggambaran keduanya menggunakan warna yang berbeda. Ketebalan kompoen 3-D yang akan digambar diabaikan, sehingga bentuk gambar berupa permukaan (*surface*) sesuai titik-titik koordinat 3-D *input*. Gambar permukaan (*surface*) pada media tampilan yang dihasilkan harus dapat mewakili bentuk benda sebenarnya. Sesuai dengan input yang berupa  $M \times N$  titik, gambar yang dihasilkan yaitu berupa *mesh* atau jaring-jaring poligon. Untuk



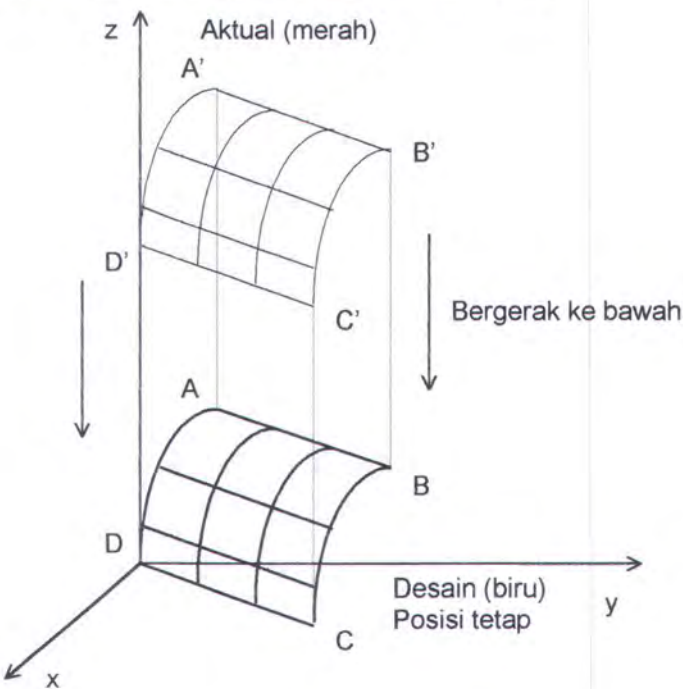
mudahnya membedakan kedua gambar yang dihasilkan, maka dibuat dengan warna yang berbeda, yaitu biru untuk desain dan merah untuk aktual. Adapun langkah penggambaran adalah sebagai berikut :

- Menggambar masing-masing input desain dan aktual.



**Gambar V-5**  
Penggambaran komponen 3-D desain dan aktual

- Menggambarkan proses *matching*, dimana gambar aktual diberi jarak tertentu lalu dijatuhkan lurus ke arah gambar komponen desain.



**Gambar V-6**  
Penggambaran proses *Matching*

### □ **Perhitungan *Matching Process***

Perhitungan dalam *matching process* menggunakan bahasa pemrograman *Visual BASIC*, yang dipakai untuk mencari besar penyimpangan. Tiap titik koordinat 3-D dihitung dengan menggunakan ketentuan bahwa besar selisih antara koordinat desain dan aktual dalam batasan toleransi yang telah ditentukan lebih dahulu dianggap tidak terjadi penyimpangan. Sedangkan pada titik-titik koordinat 3-D dengan besar selisih diluar batasan tersebut dikatakan mempunyai penyimpangan. Langkah-langkah perhitungan dalam *matching process* adalah sebagai berikut :

- Mencari selisih awal ( $\Delta x'$ ,  $\Delta y'$ ,  $\Delta z'$ ) dalam batasan toleransi; titik-titik koordinat 3-D aktual sebesar ( $x'$ ,  $y'$ ,  $z'$ ) dikurangi dengan titik-titik koordinat 3-D desain sebesar ( $x$ ,  $y$ ,  $z$ ).
- Mencari titik dengan besar selisih koordinat 3-D paling kecil, nilai masing-masing  $\Delta x'$ ,  $\Delta y'$ , dan  $\Delta z'$  dalam batasan toleransi.
- Menghitung besar penyimpangan pada semua titik yang menyimpang ( $\Delta x''$ ,  $\Delta y''$ ,  $\Delta z''$ ) ; tiap titik koordinat 3-D ( $\Delta x'$ ,  $\Delta y'$ ,  $\Delta z'$ ) dikurangi dengan ( $\Delta x'$ ,  $\Delta y'$ ,  $\Delta z'$ ) terkecil.

### □ ***Output Matching Process***

Setelah kedua data *input* baik desain maupun aktual dianalisa oleh *matching process* dalam program perhitungan, dihasilkan data *output*. *Output* tersebut ditampilkan oleh program *output* yang terdiri dari dua macam, yaitu :

#### 1. Tabel data penyimpangan

Berisikan selisih tiap titik koordinat 3-D ( $\Delta x''$ ,  $\Delta y''$ ,  $\Delta z''$ ) untuk tiap baris (M) dari 3-D *mesh* gambar komponen tersebut.



## 2. Kurva penyimpangan

Menggambarkan data penyimpangan ( $\Delta x''$ ,  $\Delta y''$ ,  $\Delta z''$ ) tiap baris (M) untuk masing-masing koordinat x, koordinat y, dan koordinat z. Dengan menggunakan kurva ini kita dapat melihat penyimpangan yang terjadi sepanjang potongan M dan N dari komponen tersebut.



## V.5. BAHASA PEMROGRAMAN *MATCHING PROCESS*

Bahasa pemrograman yang digunakan dalam penyusunan program *matching process* adalah bahasa pemrograman tingkat tinggi dengan keuntungan sebagai berikut :

- mudah dipelajari dan dimengerti
- waktu penyusunan program relatif singkat
- kemungkinan terjadinya kesalahan relatif kecil, sehingga koreksi atau revisi yang dilaksanakan sedikit
- dapat dipakai pada semua aplikasi komputer

Tetapi selain itu masih mempunyai satu kelemahan utama yaitu dalam menerjemahkan kedalam bahasa mesin membutuhkan waktu yang relatif lama dibandingkan bahasa *assembler*.

Oleh karena itu dipilihlah dua bahasa pemrograman tingkat tinggi yaitu *LISP* dan *BASIC*. *LISP* termasuk dalam bahasa tingkat tinggi *problem oriented*, yaitu bahasa pemrograman dengan tujuan khusus yang digunakan untuk menyelesaikan data yang khusus pula. Dalam program *matching process* digunakan untuk bagian proses penggambaran grafis. Sedangkan *BASIC* termasuk dalam jenis bahasa tingkat tinggi *procedure oriented*, yaitu menyatakan

prosedur atau logika pemrosesan data. Dalam program *matching process* digunakan untuk program utama dan proses perhitungan.

#### V.5.1. *AutoLISP*

*AutoCAD* ditulis dengan menggunakan bahasa C, dimana sangat berbeda dengan *LISP*. *LISP* merupakan persamaan nama dari *LISt Processing*, yang menandakan bahwa *LISP* bekerja dengan menggunakan daftar (*list*) yang masuk dalam kategori bahasa tingkat tinggi *problem oriented*. *AutoLISP* adalah bahasa tambahan dalam paket ADE-3 *AutoCAD*. *AutoLISP* memungkinkan pemakai untuk mengerjakan model matematika menengah dari mode *Editor* (bersamaan pengerjaan suatu gambar) dan untuk menulis program yang digunakan dalam perhitungan serta mengatur perintah-perintah *AutoCAD*. Program-program dapat berupa bagian dari menu makro, pernyataan *LISP*, atau fungsi-fungsi yang ditentukan oleh pemakai melalui file *menu* atau file *script*.

#### □ Cara Membuat dan Menjalankan Program *AutoLISP*

Untuk menjalankan program *AutoLISP* hanya diperlukan *software AutoCAD*, tidak perlu *software* lainnya. Sedangkan untuk menulis program *AutoLISP* digunakan *software wordprocessor* seperti *wordstar*, *edlin* atau *software wordprocessor* lainnya. Ada dua macam cara menjalankan program *AutoLISP*, yaitu :

1. Menuliskan semua perintah yang dijalankan secara langsung lewat *prompt Command AutoCAD*.
2. Dengan membuat program yang disimpan dalam file dengan *extention .lsp* kemudian baru dipanggil untuk dijalankan.



#### □ **Memanggil File Program *AutoLISP***

Perintah *LOAD* digunakan untuk memanggil suatu file program *AutoLISP* dan menjalankan program tersebut. Cara yang digunakan dengan mengetikkan perintah *LOAD* pada *prompt Command AutoCAD* diikuti dengan nama file program *AutoLISP* dan diakhiri dengan menekan tombol *Enter*. Jika dalam operasi *LOAD* terjadi kegagalan biasanya disebabkan oleh kesalahan pemrograman *AutoLISP*.

#### **V.5.2. *Visual BASIC***

*Visual Basic* adalah suatu bahasa pemrograman dalam struktur *Windows* yang mengkombinasikan kemampuan bahasa *BASIC* dengan peranti desain visual. Bahasa ini menyediakan kemudahan dan kesederhanaan pemakaian tanpa mengorbankan kinerja atau fasilitas grafis yang menyebabkan *Windows* menjadi lingkungan kerja yang menyenangkan. Perancangan komponen-komponen dalam struktur *Windows* ; menu, font, kotak dialog, field teks gulung, dsb, dapat dengan mudah dibuat.

*Visual Basic* juga merupakan bahasa komputer yang mendukung pemrograman *event-driven*. *Event-driven* adalah gaya pemrograman yang cocok digunakan untuk antarmuka (*interface*) pemakai grafis. Pemrograman *event-driven* menuliskan suatu program yang bereaksi terhadap tindakan pemakai; memilih sebuah perintah, mengklik jendela, menggeser mouse, dsb, sebagai ganti penulisan sebuah program dengan mengeplot setiap langkah yang ada pada prosedur pemrograman biasa. Sebagai ganti program yang besar, dengan *Visual Basic* dapat dibuat aplikasi yang sebenarnya merupakan kumpulan miniprogram yang didpicu oleh *event-event* yang diaktifkan secara cepat dan mudah.

Keuntungan penggunaan bahasa *Visual BASIC* adalah sebagai berikut :

- Penyusunan suatu program komputer menjadi lebih sederhana.
- Merupakan bahasa pemrograman yang menggunakan sistem operasi *Windows* sehingga dapat digunakan untuk pembuatan fungsi makro dari program-program aplikasi *Windows*.

Hasil dari uraian diatas adalah suatu program komputer yang merupakan prototipe program *matching process*. Uraian secara terperinci mengenai prototipe program ini akan dijelaskan pada bab berikutnya.



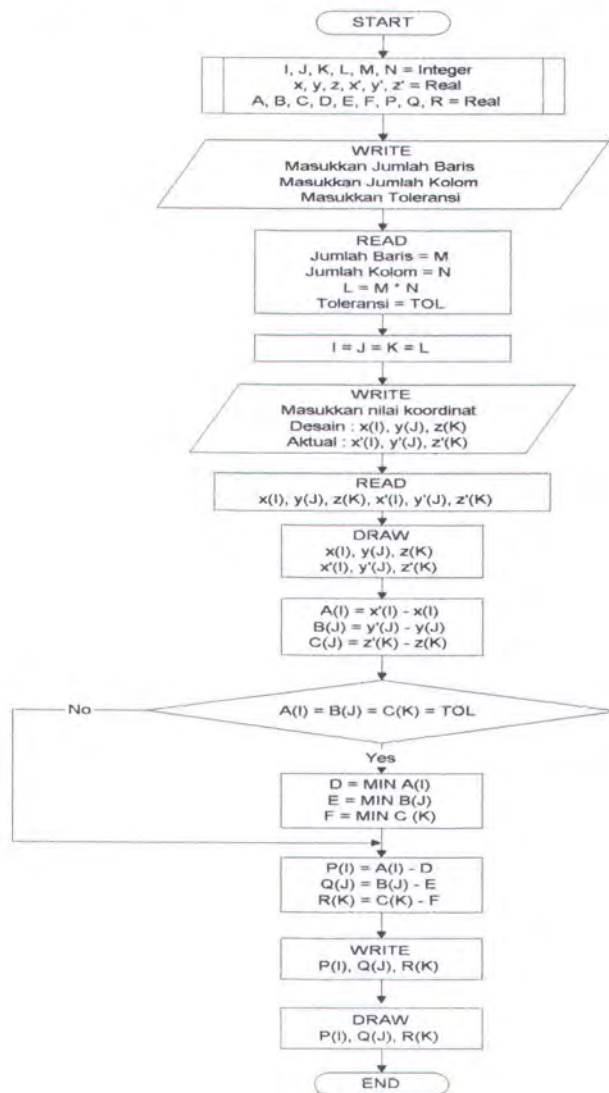
***BAB VI***  
***PROTOTYPE PROGRAM***  
***MATCHING PROCESS***

## BAB VI

### PROTOTIPE PROGRAM MATCHING PROCESS

#### VI.1. FLOWCHART PROGRAM MATCHING PROCESS

Berikut ini merupakan *flowchart matching process* yang nantinya digunakan sebagai dasar atau pedoman pembuatan program. *Flowchart* ini juga menggambarkan logika perhitungan dan alur proses program :



**Gambar VI-1**  
*Flowchart program Matching Process*



## V2. STRUKTUR PROGRAM *MATCHING PROCESS*

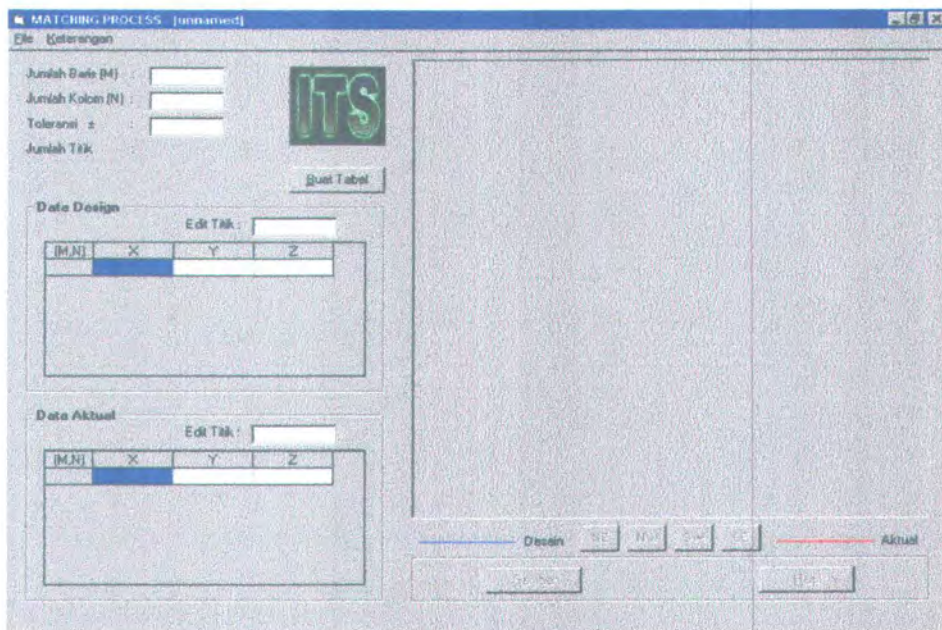
Dalam *Visual BASIC* struktur program disusun dalam suatu bentuk *form* yang membentuk suatu tampilan *window* (jendela). Satu *form* membentuk satu tampilan *window*. Dalam suatu *form* dapat disusun suatu *listing* program sesuai dengan kebutuhan yang diperlukan (*listing* program *matching process* terdapat pada lampiran D). Struktur prototipe program *matching process* ini disusun menjadi beberapa *form* yaitu :

1. *Main.Frm* ; berisikan program utama tentang pembacaan data *input* yang dibagi menjadi beberapa sub program, yaitu :
  - *Mnufilebaru* ; berfungsi memulai file baru.
  - *Bukafile* ; berfungsi membuka sebuah file.
  - *Simpanfile* ; berfungsi meyimpan file.
  - *Btabel* ; berfungsi membuat tabel "Data Desain" dan "Data Aktual".
  - *Filetx1* ; berfungsi menyimpan data input dalam file "*matching.txt*" yang akan digunakan untuk data penggambaran. Terdiri dari dua sub program yaitu *filetxt1* dan *filetxt2*.
  - *Gambar* ; berfungsi mengaktifkan program *AutoCAD* dan menampilkan gambar.
  - *Hitung* ; menghitung data penyimpangan.
  - *Kotak3* ; berfungsi membuat tabel "Data Penyimpangan"
  - *Kurva* ; berfungsi menampilkan kurva penyimpangan.
2. *Hasil.Frm* ; berisikan sub program perhitungan dan tampilan tabel "Data Penyimpangan".
3. *KurvMX.Frm*, *KurvaMY.Frm*, dan *KurvaMZ.Frm* ; berisi sub program pembuatan kurva penyimpangan pada potongan M berikut tampilannya.

4. *KurvNX.Frm*, *KurvaNY.Frm*, dan *KurvaNZ.Frm* ; berisi sub program pembuatan kurva penyimpangan pada potongan N berikut tampilannya.
5. *Ktr.Frm* ; berisi sub program keterangan.

### VI.2.1. MAIN.FRM

Merupakan *window* utama yang ditampilkan pertama kali saat program *matching process* dijalankan (*Gambar VI-2*).



**Gambar VI-2**  
Tampilan *Main.Frm*

*Main.Frm* ini terdiri dari :

- *Menu bar* :
  - File : file baru, buka file, simpan file, dan keluar
  - Keterangan
- *Text* "Jumlah Baris (M)", "Jumlah Kolom (N)", "Toleransi", dan "Jumlah Titik"
- *Command button* "Buat Tabel"
- *Grid* tabel "Data Desain"



- *Grid* tabel “Data Desain”
- *Grid* tabel “Data Aktual”
- *Command button* “Gambar”
- *Image* gambar komponen 3-D
- *Command button* untuk pandangan isometri : “NE”, “NW”, “SW”, “SE”
- *Command button* “Hasil”

Langkah penggunaan :

- Memasukkan jumlah baris
- Memasukkan jumlah kolom
- Memasukkan besar toleransi
- Tekan tombol gambar untuk melihat bentuk grafis komponen 3-D kapal.
- Untuk merubah pandangan isometri digunakan tombol “NE”, “NW”, “SW”, dan “SE”
- Tekan tombol hasil untuk mendapatkan hasil perhitungan atau memunculkan *window* “Hasil”

### VI.2.2. HASIL.FRM

Merupakan *window* yang akan ditampilkan jika *command button* “Hasil” pada *Main.Frm* ditekan (*Gambar VI-3*). Pada *window* ini dapat dilihat hasil perhitungan yaitu data penyimpangan untuk titik X, Y, dan Z.

The screenshot shows a window titled "Hasil" with three data entry tables on the left and two groups of buttons on the right.

**Data Penyimpangan X**

M \ N	0	1
0	0.000	0.000
1	0.000	0.000

**Data Penyimpangan Y**

M \ N	0	1
0	0.000	0.000
1	0.000	0.000

**Kurva Penyimpangan Z**

M \ N	0	1
0	0.000	0.000
1	0.000	0.000

**Kurva Penyimpangan Potongan M**

- Kurva Penyimpangan X
- Kurva Penyimpangan Y
- Kurva Penyimpangan Z

**Kurva Penyimpangan Potongan N**

- Kurva Penyimpangan X
- Kurva Penyimpangan Y
- Kurva Penyimpangan Z

Ok

**Gambar VI-3**  
Tampilan Hasil.Frm

Hasil.Frm ini terdiri dari :

- Grid tabel "Data Penyimpangan X"
- Grid tabel "Data Penyimpangan Y"
- Grid tabel "Data Penyimpangan Z"
- Command button Kurva Penyimpangan Potongan M : "Kurva Penyimpangan X", "Kurva Penyimpangan Y", "Kurva Penyimpangan Z".
- Command button Kurva Penyimpangan Potongan N : "Kurva Penyimpangan X", "Kurva Penyimpangan Y", "Kurva Penyimpangan Z".
- Command button "Ok"

### VI.2.3. KURVAMX.FRM, KURVAMY.FRM, KURVAMZ.FRM

Merupakan windows yang akan ditampilkan jika command button "Kurva Penyimpangan X", "Kurva Penyimpangan Y", dan "Kurva Penyimpangan Z" pada



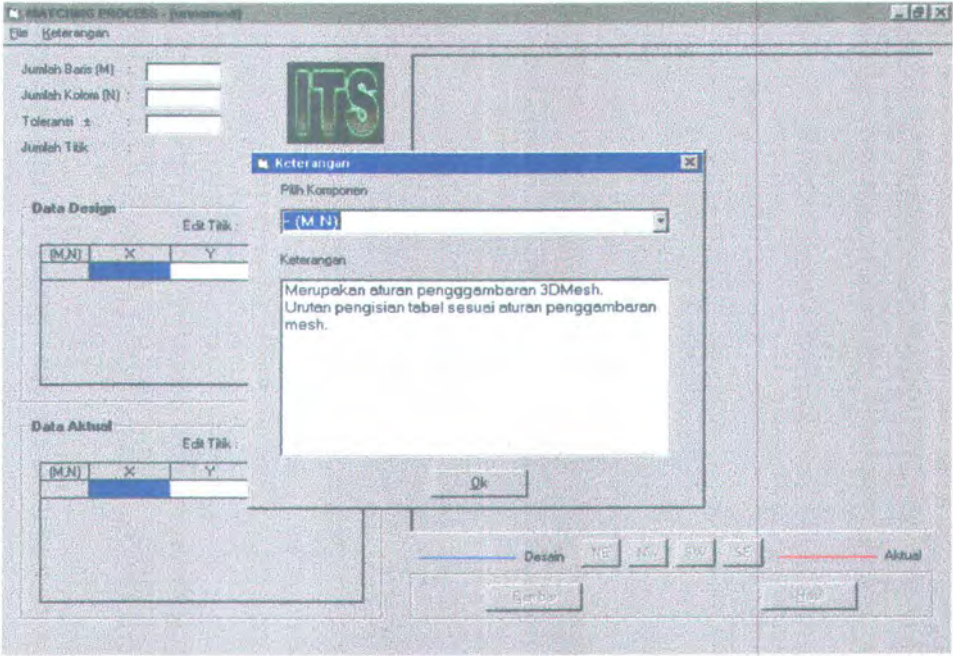
Hasil.Frm untuk potongan M ditekan. *Window* ini berisikan gambar kurva penyimpangan untuk potongan M.

**VI.2.4. KURVANX.FRM, KURVANY.FRM, KURVANZ.FRM**

Merupakan *windows* yang akan ditampilkan jika *command button* “Kurva Penyimpangan X”, “Kurva Penyimpangan Y”, dan “Kurva Penyimpangan Z” pada Hasil.Frm untuk potongan N ditekan. *Window* ini berisikan gambar kurva penyimpangan untuk potongan N.

**VI.2.5. KTR.FRM**

Merupakan *window* yang akan ditampilkan jika menu “Keterangan” pada *menu bar* pada Hasil.Frm dipilih (Gambar VI-4). *Window* sebagai pendukung yang berisikan keterangan mengenai komponen-komponen program.



**Gambar VI-4**  
Tampilan Ktr.Frm

Ktr.Frm ini terdiri dari :

- *Combo* "Pilih Komponen"
- *List* "Keterangan"
- *Command button* "Ok"

### VI.3. *RUNNING PROGRAM*

Percobaan *running program matching* ini menggunakan komponen yang sudah ditentukan pada Bab V.1. yaitu pelat lengkung tidak simetris (No. A140425).

Data-data yang dimasukkan adalah :

- Jumlah baris (M) = 4
- Jumlah kolom (N) = 12
- Toleransi =2.5 mm
- Data desain, diperoleh dari pembacaan gambar desain (dalam mm) adalah :

(M,N)	X	Y	Z
(0,0)	0	0	0
(0,1)	150	25	-17.5
(0,2)	750	90	-60
(0,3)	1350	145	-100
(0,4)	1950	190	-140
(0,5)	2550	227.5	-180
(0,6)	3150	257.5	-215
(0,7)	3750	280	-250
(0,8)	4350	300	-275
(0,9)	4950	317.5	-300
(0,10)	5550	330	-315
(0,11)	5730	335	-320
(1,0)	0	215	110
(1,1)	150	240	95
(1,2)	750	325	50
(1,3)	1350	405	5
(1,4)	1950	482.5	-35
(1,5)	2550	550	-75
(1,6)	3150	610	-115
(1,7)	3750	665	-150
(1,8)	4350	720	-185



(1,9)	4950	765	-220
(1,10)	5550	800	-250
(1,11)	5730	815	-265
(2,0)	0	420	225
(2,1)	150	450	220
(2,2)	750	557.5	170
(2,3)	1350	661.25	130
(2,4)	1950	765	90
(2,5)	2550	865	52.5
(2,6)	3150	957.5	17.5
(2,7)	3750	1047.5	-25
(2,8)	4350	1130	-60
(2,9)	4950	1205	-100
(2,10)	5550	1270	-135
(2,11)	5730	1290	-160
(3,0)	0	610	350
(3,1)	150	640	340
(3,2)	750	775	300
(3,3)	1350	910	265
(3,4)	1950	1037.5	230
(3,5)	2550	1160	197.5
(3,6)	3150	1280	165
(3,7)	3750	1400	130
(3,8)	4350	1520	100
(3,9)	4950	1635	70
(3,10)	5550	1720	20
(3,11)	5730	1750	-10

**Tabel VI-1**  
Data Desain (Titik Koordinat 3-D)

- Data aktual, hasil pengukuran dengan *marking table* (dalam mm) adalah :

(M,N)	X	Y	Z
(0,0)	0	0	0
(0,1)	149.135	24.685	-18.155
(0,2)	749.135	90.165	-58.695
(0,3)	1349.135	144.985	-99.365
(0,4)	1949.135	188.35	-142.55
(0,5)	2549.135	228.545	-179.56
(0,6)	3149.135	257.24	-218.585
(0,7)	3749.135	279.755	-253.66
(0,8)	4349.135	299.215	-278.145
(0,9)	4949.135	319.465	-299.365
(0,10)	5549.135	332.15	-317.85
(0,11)	5728.09	337.04	-320.145
(1,0)	0	214.565	109.455

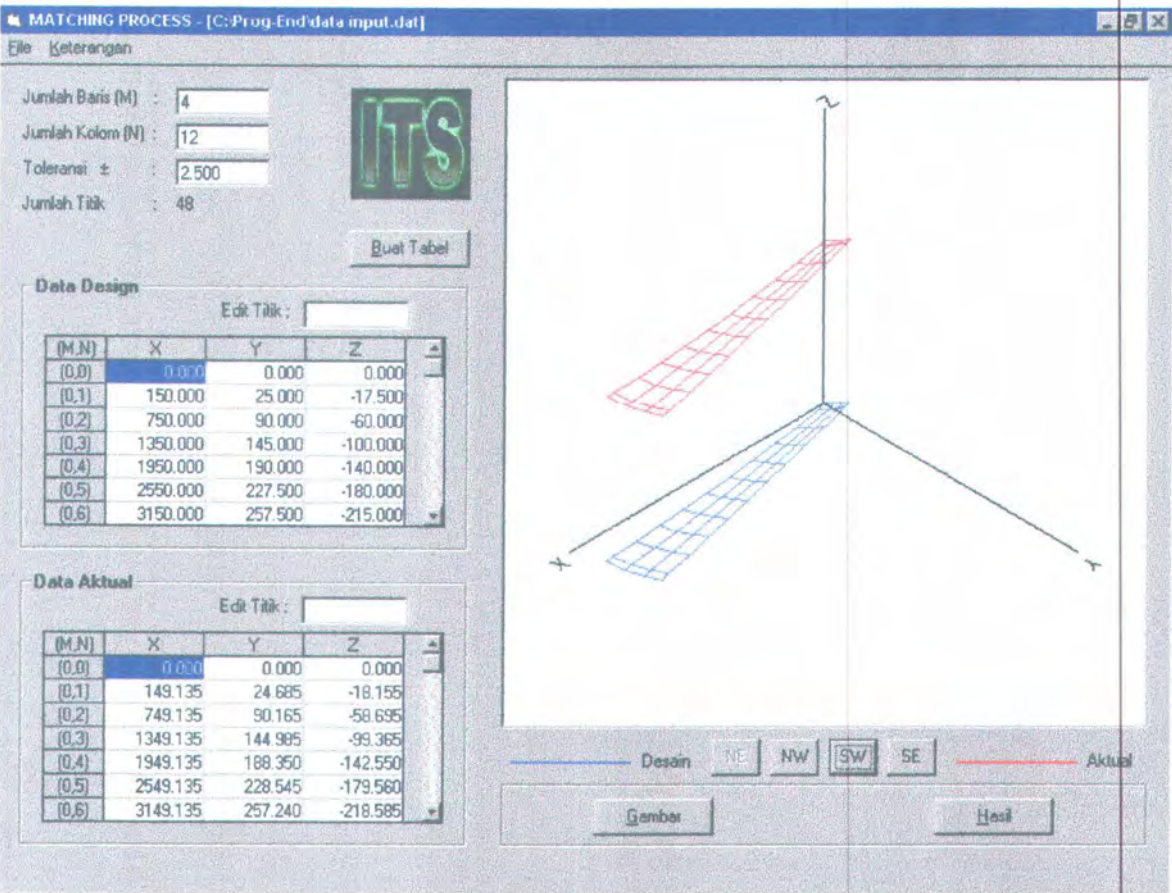
(1,1)	149.135	240.055	94.57
(1,2)	749.135	324.155	49.78
(1,3)	1349.135	407.13	4.155
(1,4)	1949.135	483.145	-34.155
(1,5)	2549.135	550.985	-76.98
(1,6)	3149.135	612.58	-114.285
(1,7)	3749.135	665.485	-152.16
(1,8)	4349.135	719.15	-184.035
(1,9)	4949.135	762.16	-218.165
(1,10)	5549.135	801.54	-247.155
(1,11)	5728.09	814.98	-268.69
(2,0)	0	421.37	226.99
(2,1)	149.135	450.33	224.13
(2,2)	749.135	558.69	169.165
(2,3)	1349.135	659.475	127.995
(2,4)	1949.135	764.135	91.69
(2,5)	2549.135	864.98	53.115
(2,6)	3149.135	958.16	18.46
(2,7)	3749.135	1048.66	-25.66
(2,8)	4349.135	1129.65	-59.655
(2,9)	4949.135	1208.245	-102.36
(2,10)	5549.135	1269.995	-136.25
(2,11)	5728.09	1289.46	-160.25
(3,0)	0	609.995	352.135
(3,1)	149.135	641.68	339.545
(3,2)	749.135	776.195	298.965
(3,3)	1349.135	914.13	267.15
(3,4)	1949.135	1038.215	231.07
(3,5)	2549.135	1161.235	198.54
(3,6)	3149.135	1280.215	165.025
(3,7)	3749.135	1401.62	130.69
(3,8)	4349.135	1521.665	100.25
(3,9)	4949.135	1635.095	71.035
(3,10)	5549.135	1721.65	20.895
(3,11)	5728.09	1749.655	-9.995

**Tabel V-2**  
Data Aktual (Titik Koordinat 3-D)

Hasil *running* program adalah :

- Penggambaran grafis





Gambar VI-5  
Penggambaran grafis komponen 3-D

Gambar warna biru merupakan gambar komponen desain sedangkan warna merah merupakan gambar komponen aktual.

▪ Data penyimpangan

Data penyimpangan yang diperoleh setelah proses perhitungan adalah sebagai berikut :

(M,N)	X	Y	Z
(0,0)	0	0	0
(0,1)	-0.865	-0.315	-0.655
(0,2)	-0.865	0.165	1.305
(0,3)	-0.865	-0.015	0.635
(0,4)	-0.865	-1.65	-2.55
(0,5)	-0.865	1.045	0.44
(0,6)	-0.865	-0.26	-3.585
(0,7)	-0.865	-0.245	-3.66

(0,8)	-0.865	-0.785	-3.145
(0,9)	-0.865	1.965	0.635
(0,10)	-0.865	2.15	-2.85
(0,11)	-1.91	2.04	-0.145
(1,0)	0	-0.435	-0.545
(1,1)	-0.865	0.055	-0.43
(1,2)	-0.865	-0.845	-0.22
(1,3)	-0.865	2.13	-0.845
(1,4)	-0.865	0.645	0.845
(1,5)	-0.865	0.985	-1.98
(1,6)	-0.865	2.58	0.715
(1,7)	-0.865	0.485	-2.16
(1,8)	-0.865	-0.85	0.965
(1,9)	-0.865	-2.84	1.835
(1,10)	-0.865	1.54	2.845
(1,11)	-1.91	-0.02	-3.69
(2,0)	0	1.37	1.99
(2,1)	-0.865	0.33	4.13
(2,2)	-0.865	1.19	-0.835
(2,3)	-0.865	-1.775	-2.005
(2,4)	-0.865	-0.865	1.69
(2,5)	-0.865	-0.02	0.615
(2,6)	-0.865	0.66	0.96
(2,7)	-0.865	1.16	-0.66
(2,8)	-0.865	-0.35	0.345
(2,9)	-0.865	3.245	-2.36
(2,10)	-0.865	-0.005	-1.25
(2,11)	-1.91	-0.54	-0.25
(3,0)	0	-0.005	2.135
(3,1)	-0.865	1.68	-0.455
(3,2)	-0.865	1.195	-1.035
(3,3)	-0.865	4.13	2.15
(3,4)	-0.865	0.715	1.07
(3,5)	-0.865	1.235	1.04
(3,6)	-0.865	0.215	0.025
(3,7)	-0.865	1.62	0.69
(3,8)	-0.865	1.665	0.25
(3,9)	-0.865	0.095	1.035
(3,10)	-0.865	1.65	0.895
(3,11)	-1.91	-0.345	0.005

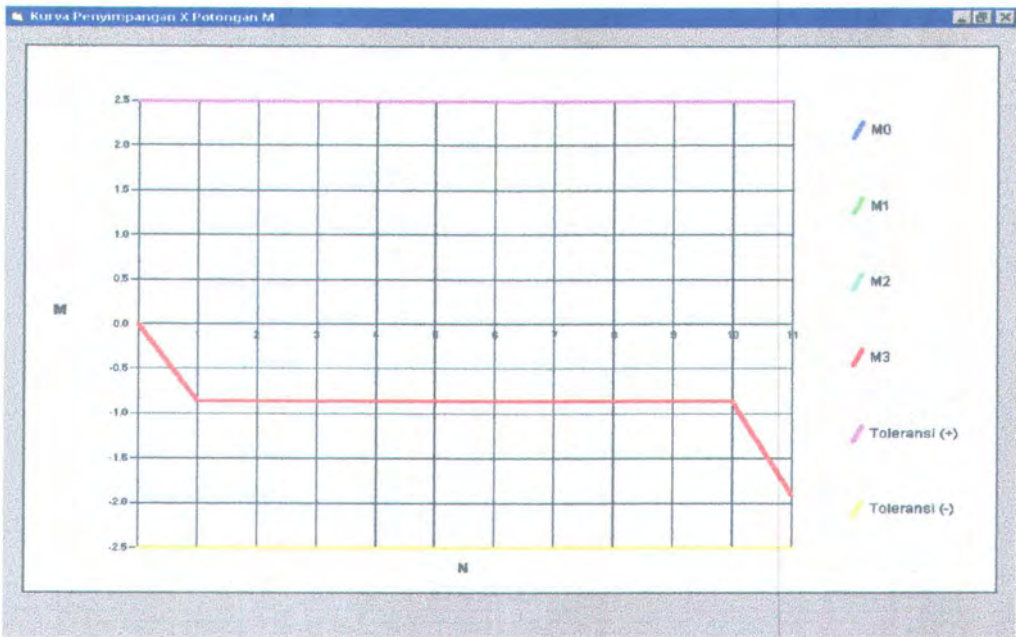
**Tabel V-3**  
Data Penyimpangan

- Grafik penyimpangan

Potongan M

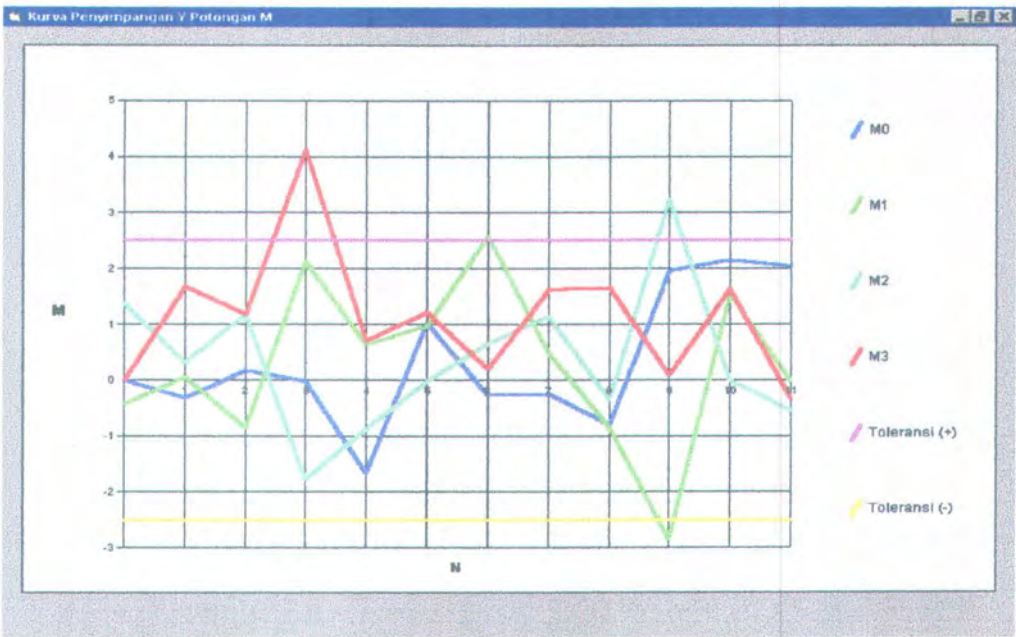


a. Kurva Penyimpangan X



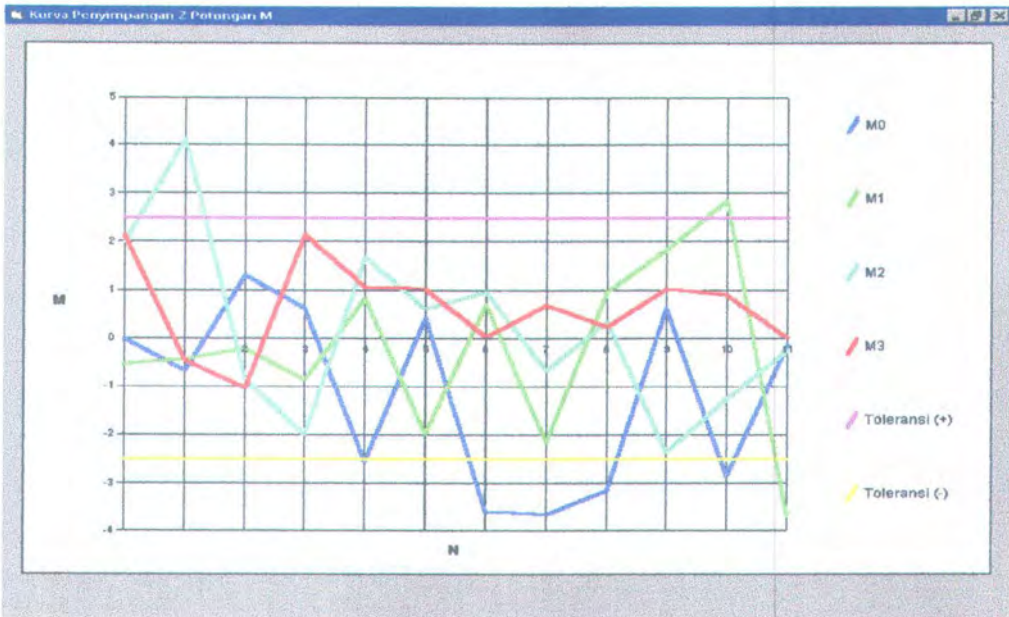
**Gambar VI-6**  
Grafik Penyimpangan X Potongan M

b. Kurva Penyimpangan Y



**Gambar VI-7**  
Grafik Penyimpangan Y Potongan M

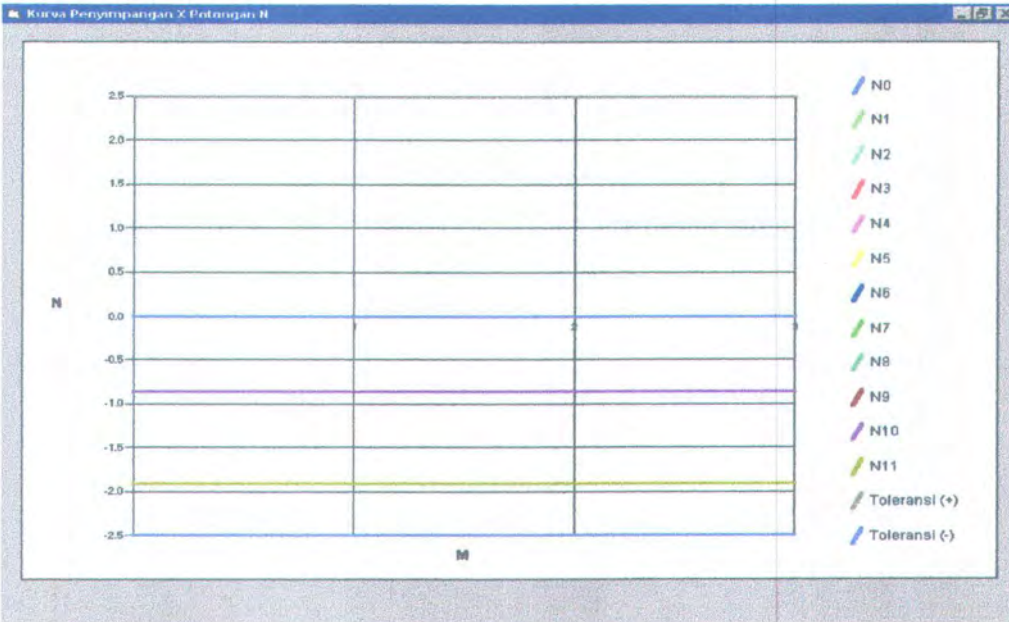
c. Kurva Penyimpangan Z



**Gambar VI-8**  
Grafik Penyimpangan Z Potongan M

Potongan M

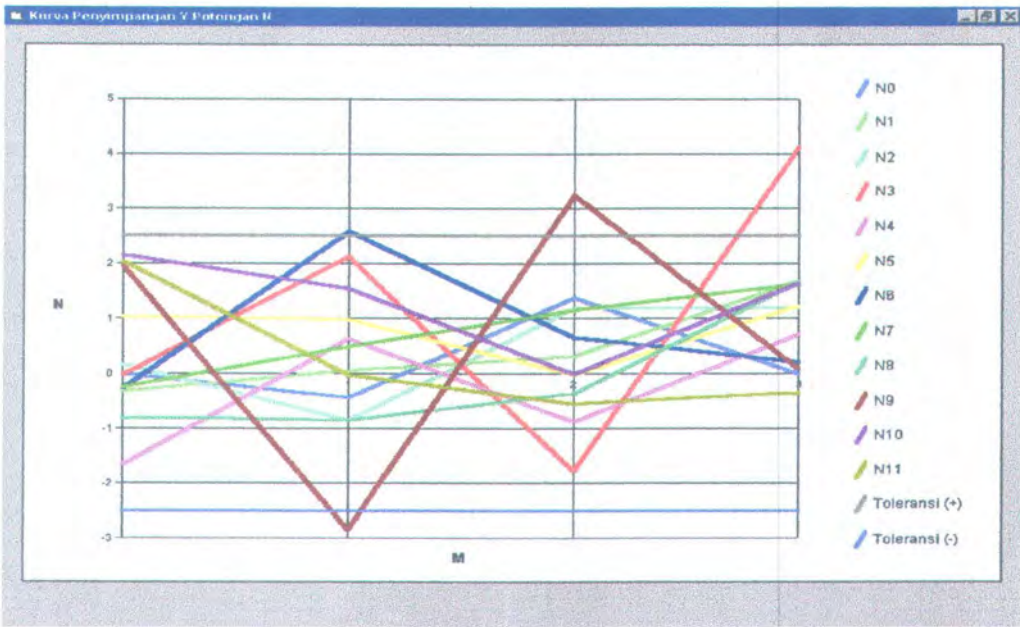
a. Kurva Penyimpangan X



**Gambar VI-9**  
Grafik Penyimpangan X Potongan N

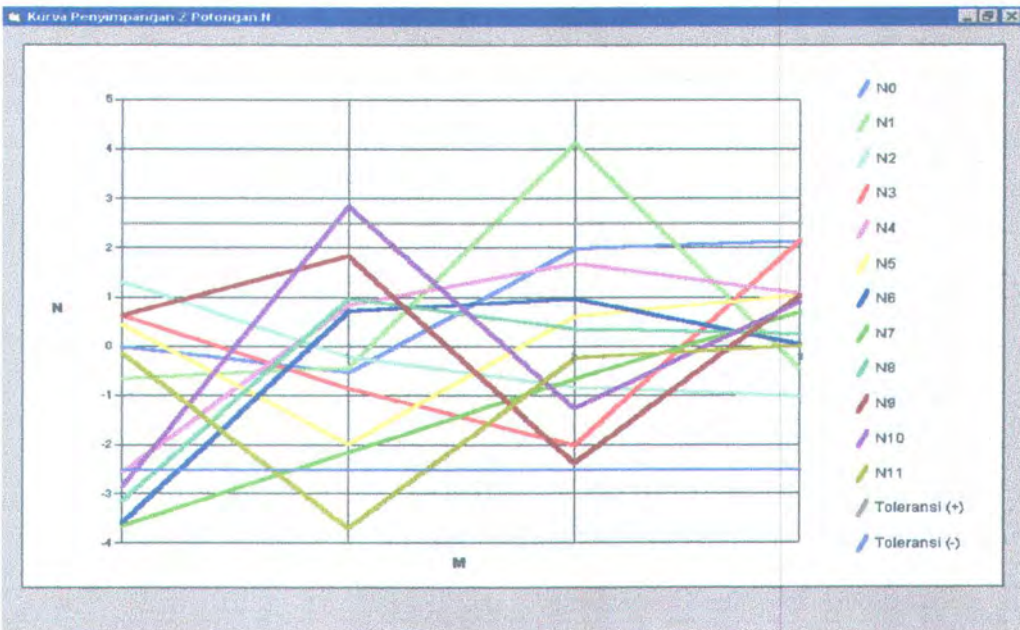


b. Kurva Penyimpangan Y



**Gambar VI-10**  
Grafik Penyimpangan Y Potongan N

c. Kurva Penyimpangan Z



**Gambar VI-11**  
Grafik Penyimpangan Z Potongan N

Dari kurva-kurva di atas kita dapat melihat apakah titik-titik pada komponen itu untuk masing-masing potongan, baik M maupun N melewati toleransi atau tidak.

**BAB VII**  
**DISKUSI DAN REKOMENDASI**



## BAB VII

### DISKUSI DAN REKOMENDASI

Dalam Tugas Akhir ini penulis mencoba menyusun suatu uraian mengenai suatu metode dalam pemeriksaan proses pembangunan kapal yang disebut *matching process* yang telah dijelaskan pada Bab-bab sebelumnya. Hal ini merupakan metode baru dan belum pernah diterapkan dalam proses pembangunan kapal maupun industri perkapalan pada umumnya. Metode ini menggunakan sistem komputerisasi dengan harapan bahwa pelaksanaan dapat lebih cepat dan hasil yang baik. Untuk sementara ini hasil dari Tugas Akhir ini berupa suatu protipe program komputer *matching process*.

Seperti telah dijelaskan pada bab-bab sebelumnya langkah metode *matching process* ini memerlukan dua data *input*, yaitu data desain dan aktual. Data desain digunakan sebagai acuan perhitungan penyimpangan yang terjadi. Sedangkan data aktual dapat diperoleh dari pengukuran komponen yang telah jadi menggunakan suatu alat tertentu, dalam hal ini penulis menggunakan *marking table*. Hasil atau data *output* program komputer *matching process* ini adalah suatu data dan kurva penyimpangan dari komponen 3-D kapal yang diperiksa. Dengan menganalisa kurva penyimpangan untuk tiap potongan dari komponen tersebut dapat diketahui besar penyimpangan yang terjadi dan dapat diketahui apakah komponen yang diproses tersebut sesuai dengan perencanaan atau tidak.

Karena metode ini benar-benar baru, hasil yang diperoleh merupakan konsep awal sehingga masih terdapat beberapa kekurangan antara lain :

- Kemampuan prototipe program yang telah jadi belum maksimal karena keterbatasan kemampuan bahasa *Visual BASIC* yang digunakan khususnya dalam penggambaran kurva penyimpangan hasil yang diperoleh masih kurang bagus.
- Data desain yang diperlukan oleh program ini sangat tergantung dari ketelitian pembacaan gambar rencana maupun gambar kerja.
- Pembacaan data aktual sangat tergantung pada kemampuan dan kondisi alat ukur yang digunakan.
- Sulitnya pembacaan data komponen kapal dalam bentuk yang kompleks dengan jumlah titik koordinat 3-D yang sangat banyak.
- Tingkat ketepatan bentuk dalam penggambaran grafis komponen tergantung pada jumlah titik koordinat yang digunakan. Hal ini dikarenakan penggambaran grafis masih menggunakan *software AutoCAD* dengan metode *3Dmesh* yang dihubungkan oleh bahasa *LISP*.

Oleh karena itu masih diperlukan beberapa penyempurnaan untuk pengembangan selanjutnya agar semua kekurangan diatas dapat dihilangkan. Saran yang dapat penulis berikan adalah :

- Dalam penyusunan program komputer sebaiknya menggunakan satu bahasa pemrograman yang mempunyai kemampuan tinggi sehingga semua langkah *matching process* baik perhitungan maupun penggambaran grafis menjadi satu kesatuan program.
- Pembacaan data desain menggunakan sistem terpadu *database* antara bagian perencanaan dan bagian produksi (*manufacturing*) agar data desain langsung dapat digunakan tanpa perlu pembacaan manual.



- Pembacaan atau pengukuran data aktual komponen aktual menggunakan suatu alat ukur yang mempunyai tingkat akurasi yang tinggi misal peralatan optik (*laser*).
- Dalam penggunaan nyata di lapangan apabila kedua cara pembacaan data *input* diatas digabungkan menjadi suatu sistem yang terpadu beserta *interface*-nya masing-masing, untuk sistem *matching* yang lebih kompleks, hasil yang diperoleh akan lebih baik dan optimal.

Selain itu kemungkinan pengembangan penggunaan *matching process* ini dapat diterapkan pada :

- Tahap *assembly* yaitu untuk mengetahui apakah pemasangan komponen-komponen menjadi seksi atau blok sudah tepat atau belum.
- Tahap *erection* yaitu untuk mengetahui apakah nantinya pemasangan antar blok sudah tepat atau belum.

Dari kedua hal tersebut diatas diperlukan pengkajian lebih lanjut mengenai bagian-bagian yang perlu diadakan proses *matching* dan posisi titik-titik acuan yang akan digunakan.

***BAB VIII***  
***KESIMPULAN***



## BAB VIII

### KESIMPULAN

Setelah melaksanakan semua langkah metode *matching process* ini hasil akhir yang diperoleh adalah :

- Tabel data penyimpangan untuk masing-masing titik koordinat (M x N) komponen 3-D kapal.
- Kurva penyimpangan untuk tiap potongan M dan potongan N.

Dari kedua data tersebut dapat dihitung penyimpangan yang terjadi pada komponen. Dengan membaca kurva penyimpangan dapat diketahui penyimpangan secara detail sepanjang potongan tiap M dan N.

Dengan demikian penggunaan sistem komputerisasi metode *matching process* pada tahap pemeriksaan besar penyimpangan yang terjadi pada proses produksi komponen kapal dapat diketahui secara detail dan akurat.

Pelaksanaan langkah data *input* desain dan aktual metode *matching process* secara manual masih dirasakan lambat dan kurang efisien karena pengerjaannya rumit dan membutuhkan waktu yang relatif lama. Dan juga sangat tergantung pada ketelitian pelaksana dalam membaca gambar dan kemampuan serta kondisi alat ukur sendiri yang digunakan mengukur komponen jadi. Untuk mengatasi kendala ini langkah data *input* dilaksanakan menggunakan sistem terpadu antara program *matching process*, *database* perencanaan dan alat ukur komponen jadi berserta *interface*-nya masing-masing digabungkan menjadi satu kesatuan alat.

## ***DAFTAR PUSTAKA***



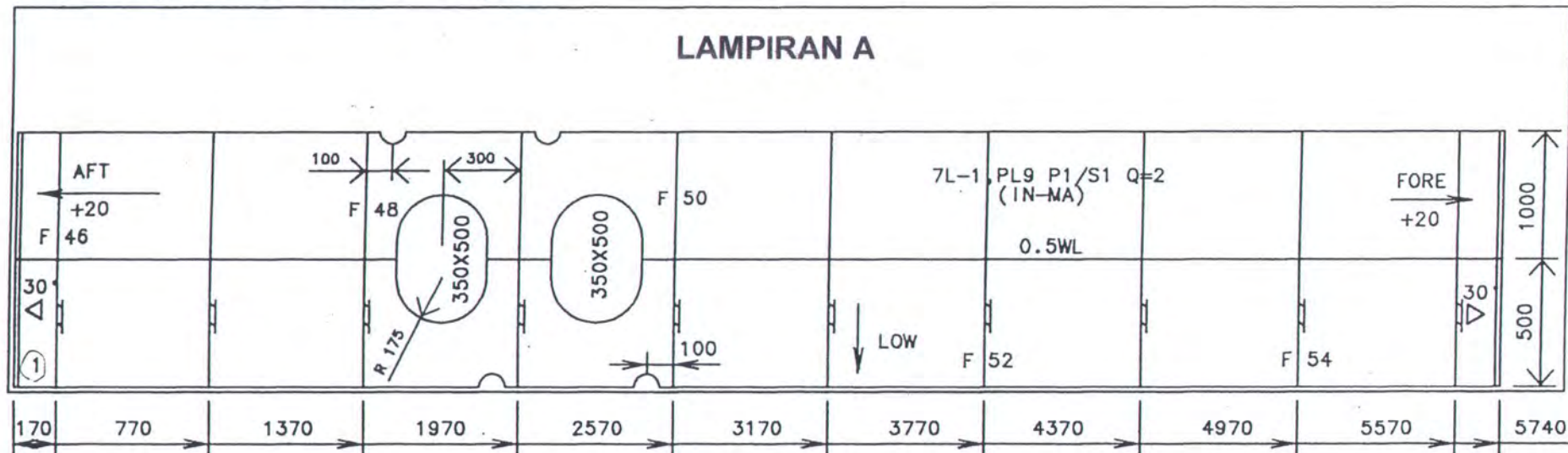
## DAFTAR PUSTAKA

- Associate Professor Ola Westby, **Dimensional Control for Building a Ship-a key topic of Managing Ship Production**, University of Strathclyde, Glasgow.
- Beach, Robert C., **An Introduction to the Curves and Surfaces of Computer-Aided Design**, Van Nostrand Reinhold, New York, 1991.
- Djoko Purwanto , Mufid, **Pedoman Pemakaian AutoLisp**, Andi Offset, Yogyakarta, 1993.
- Hearn, Donald dan Pauline Baker, M., **Computer Graphics 2<sup>nd</sup> Edition**, Prentice Hall, Inc., New Jersey, 1994.
- Manninen, Markku dan Jaatinen, Jarl, **Productive Method and System to Control Dimensional Uncertainties at Final Assembly Stages in Ship Production**, *Journal of Ship Production Vol. 8 No. 4*, November 1992.
- Medland, A.J. dan Mullineux, G., **Principles of CAD**, Kogan Page Ltd, London, 1988.
- Omura, George, **Mastering AutoCAD Release 13**, Sybex.
- Sjartuni, Ananta, **Visual Basic 4.0 dan Akses Basis Data**, PT Elex Media Komputindo, Jakarta, 1996.
- Soejitno, **Diktat Teknik Produksi Kapal**, FTK ITS.
- Yuzaki, Masaaki dan Okomoto, Yasuhisa, **An Approach to a New Ship Production System Based on Advanced Accuracy Control**, *Journal of Ship Production Vol. 9 No. 2*, May 1993.

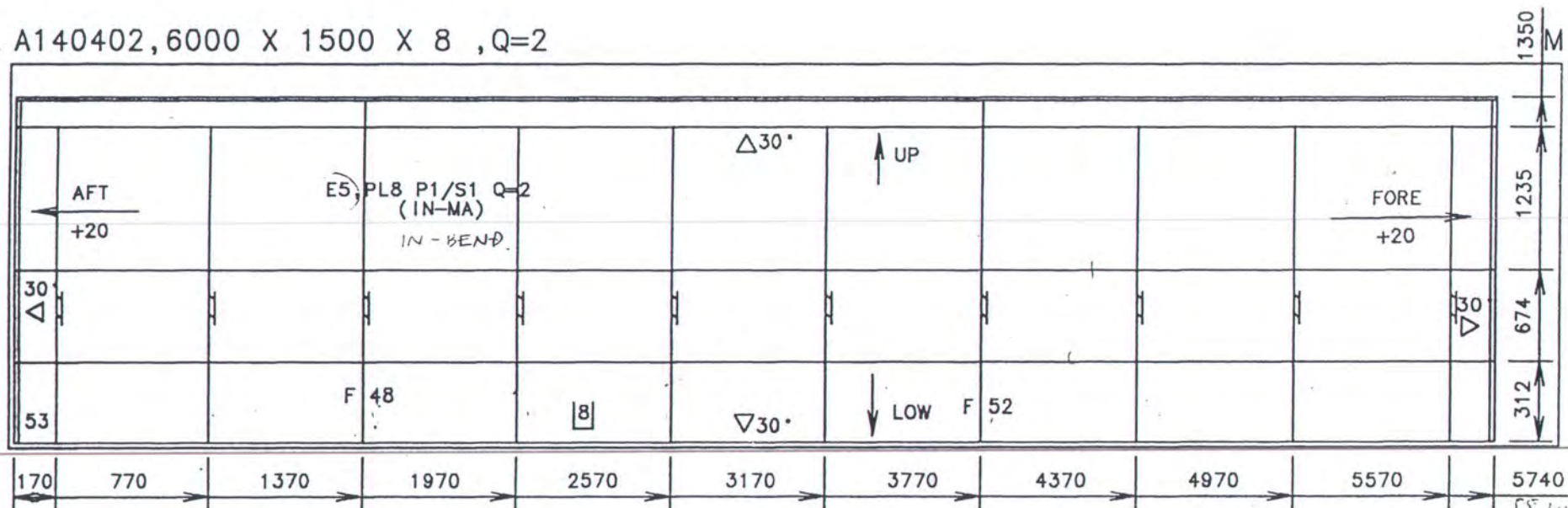
***LAMPIRAN***



A140401, 6000 X 1500 X 9 , Q=2

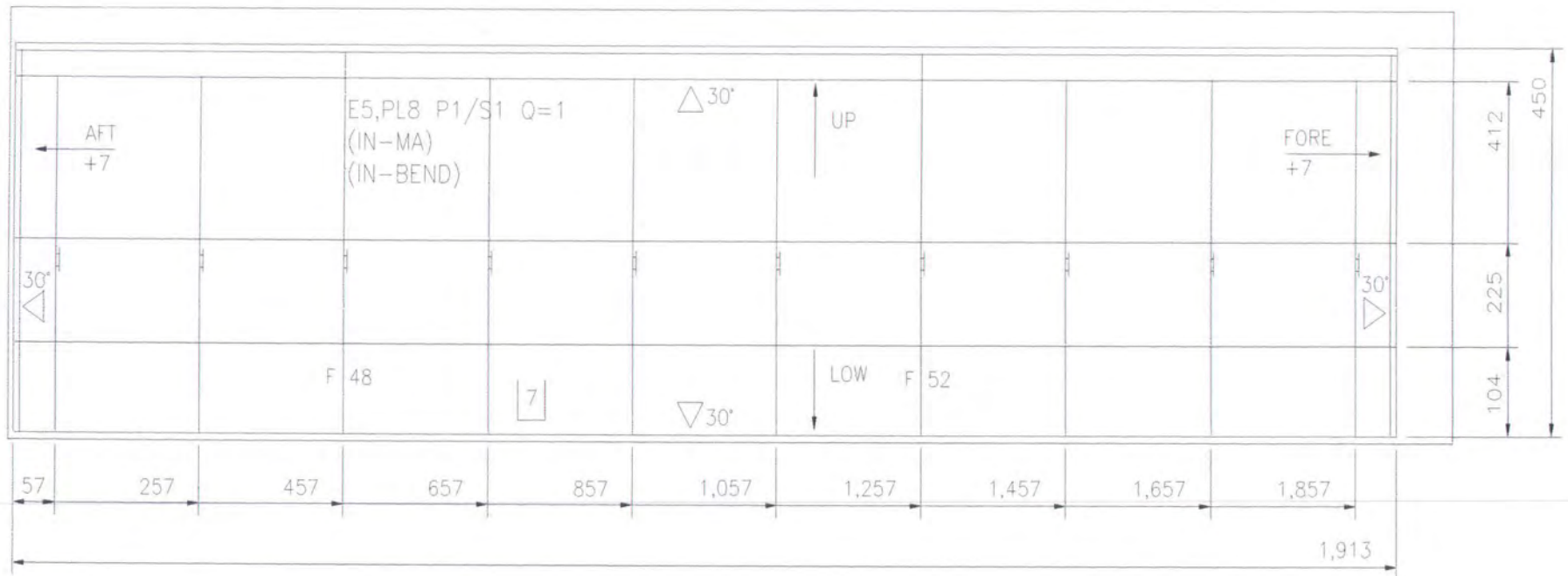


A140402, 6000 X 1500 X 8 , Q=2



Gambar Kerja Pelat Lengkung Simetris

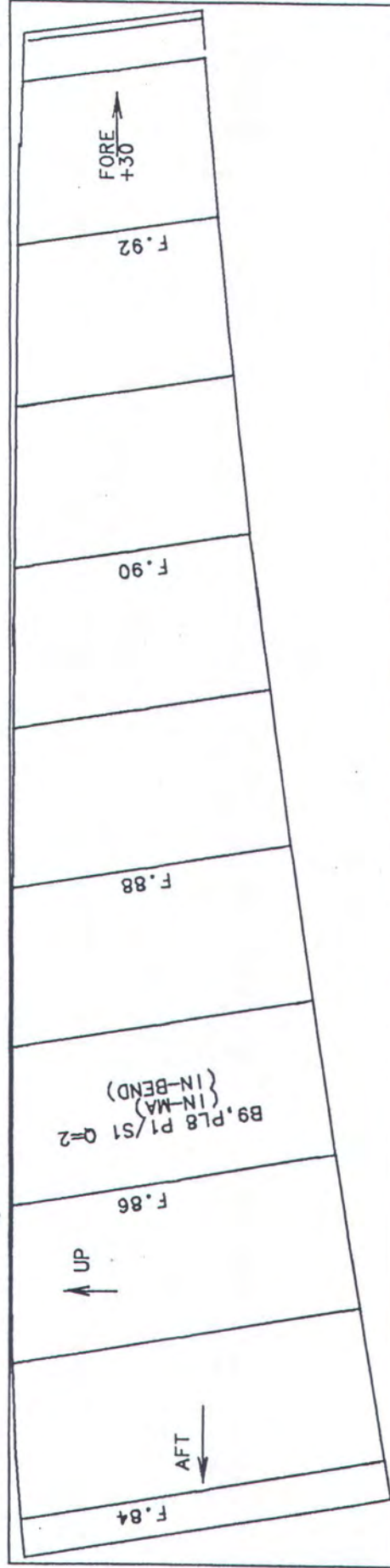
A140402, 2000X500X7, Q=1





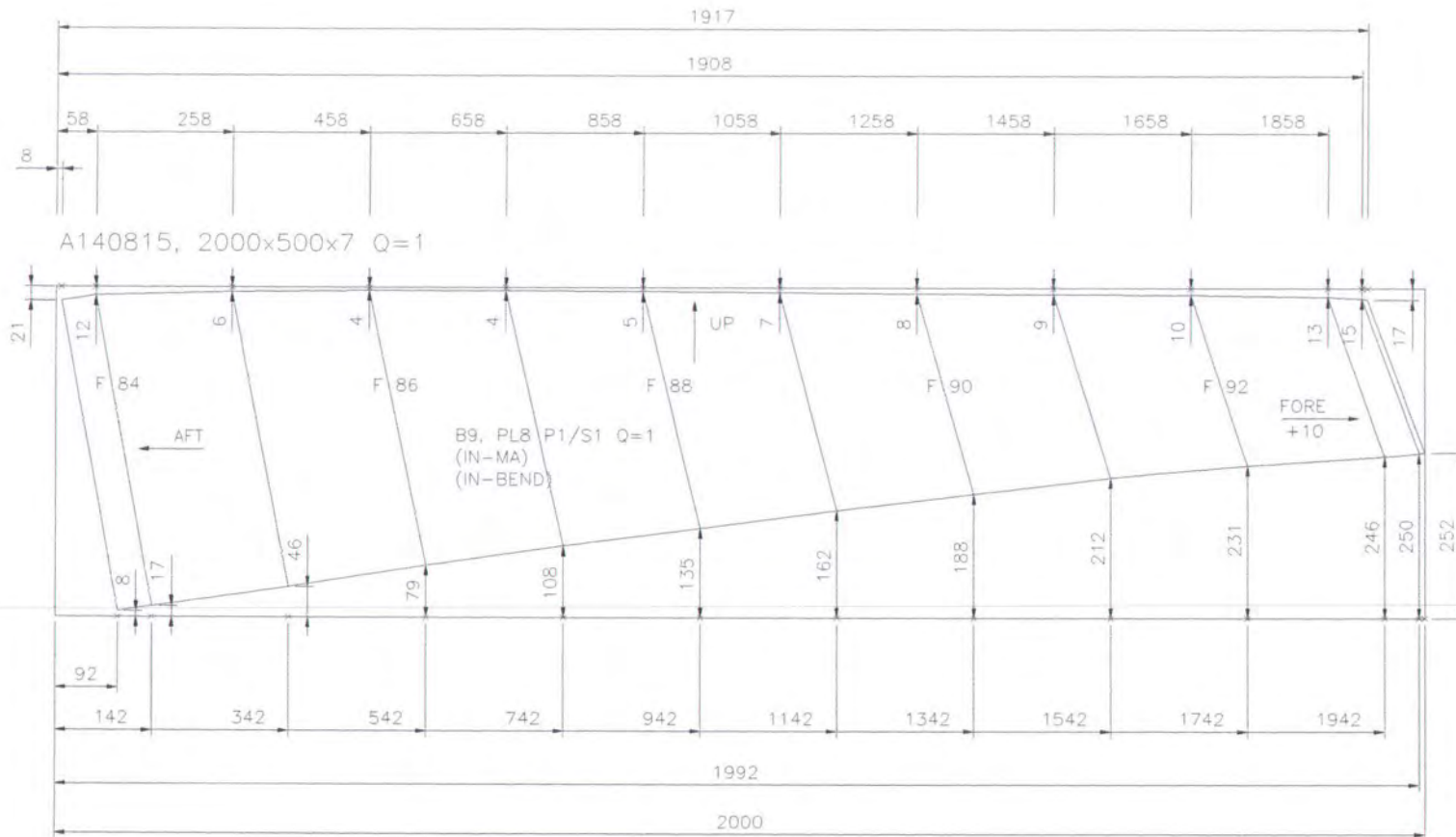
A140815, 6000X1500X8 Q=2

NC

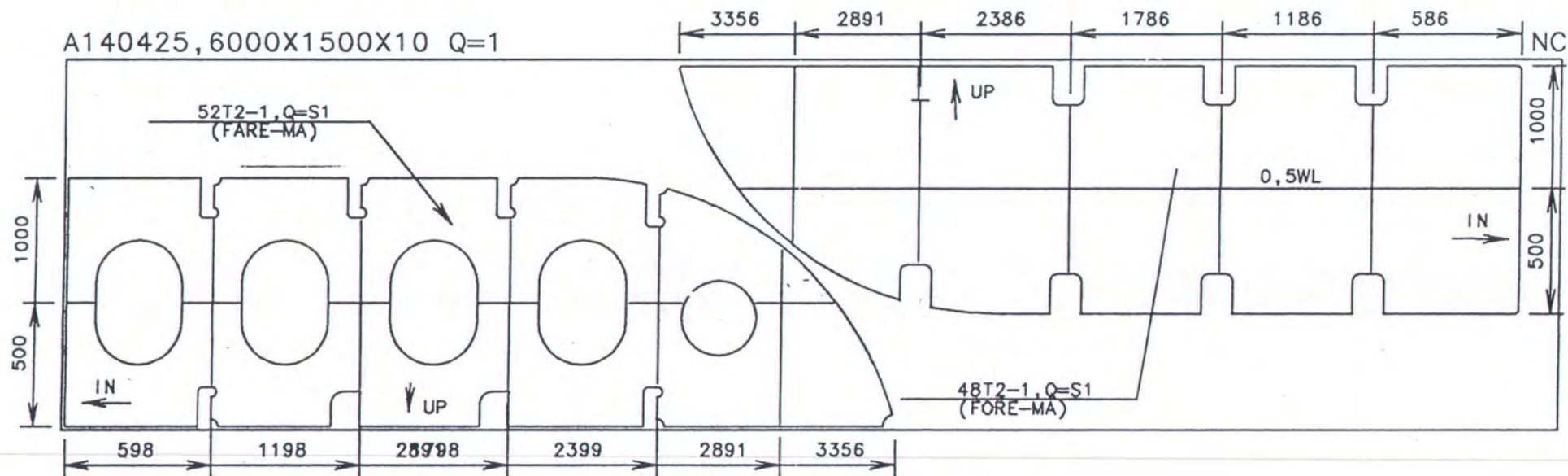


2.5

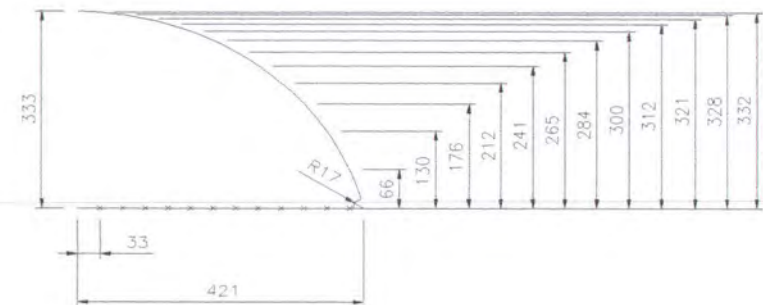
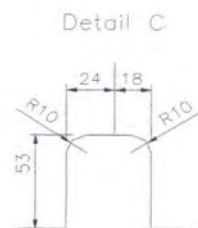
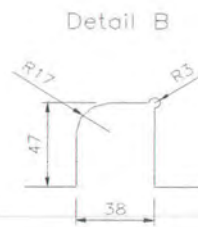
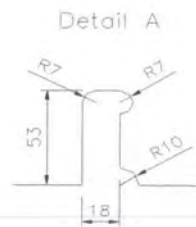
Gambar Kerja Pelat Lengkung Tidak Simetris





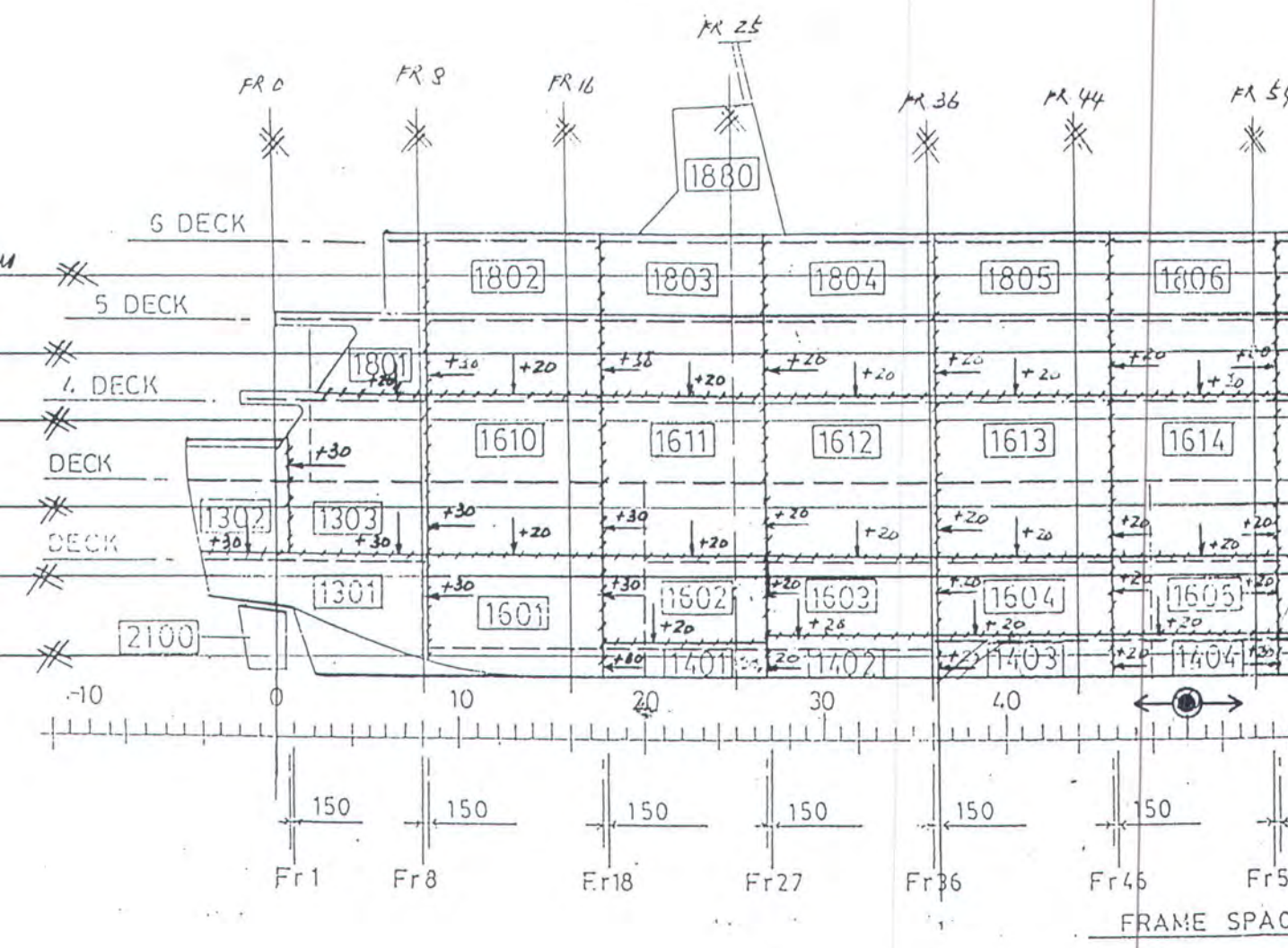


### Gambar Kerja Inner Plate

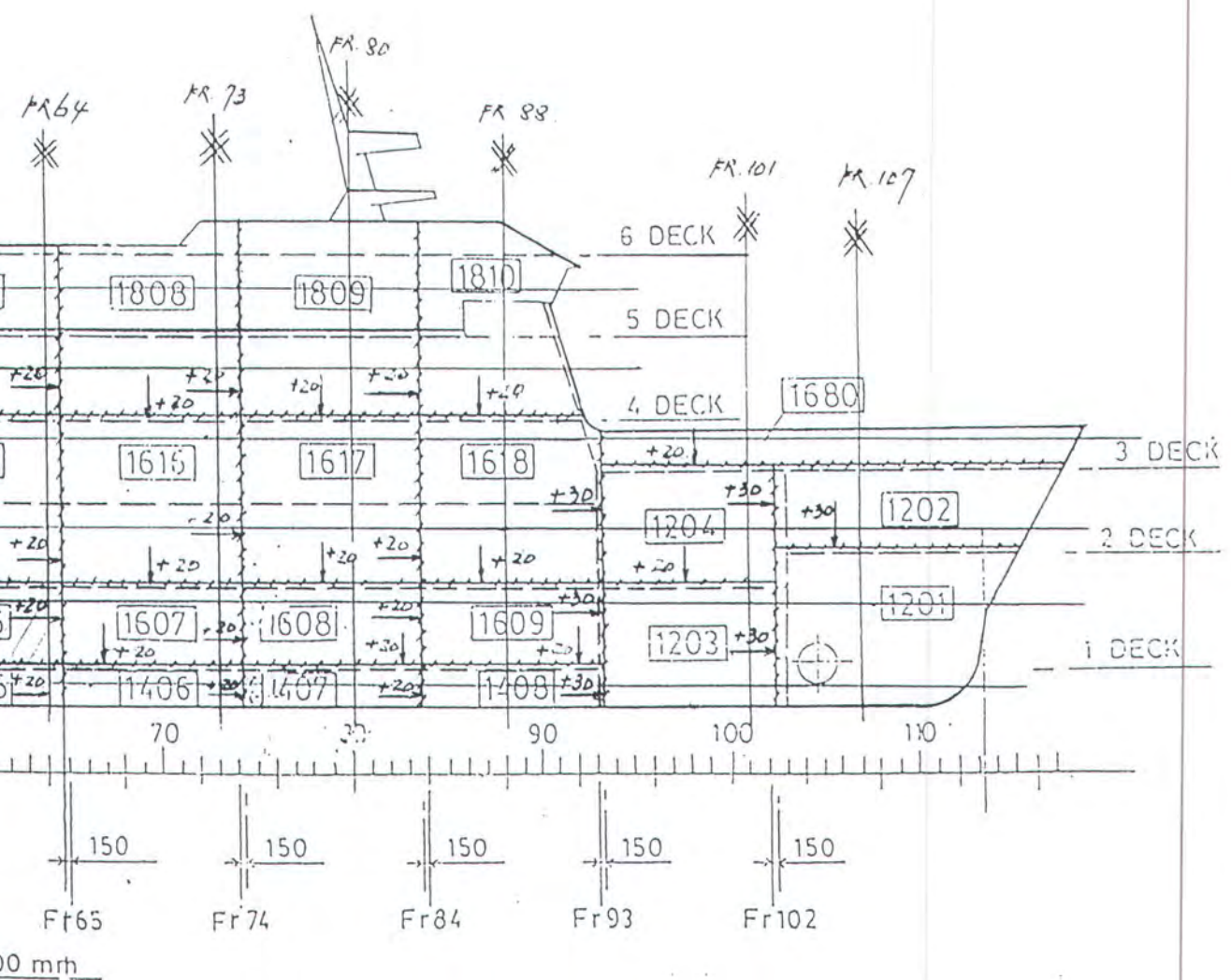




# BLOCK DIVISION PASSANGER F



→ : MARGIN PLATE  
 ※—※ : STANDART LINE





## LAMPIRAN B

### PEMBACAAN DATA DESAIN

Pelaksanaan pembacaan data desain memerlukan beberapa gambar yaitu; gambar kerja, *body plan*, dan bukaan kulit. Adapun cara-cara pembacaan data desain adalah sebagai berikut :

#### ❑ Pelat Datar (Panel)

##### ▪ Letak vertikal

Pembacaan data desain pelat datar vertikal :

- Menentukan titik origin pada ujung pelat dengan posisi paling dekat dari *base line* dan paling dekat dari AP.
- Nilai koordinat y semua titik adalah nol.
- Pembacaan nilai koordinat x sesuai letak titik sepanjang sumbu x (memanjang kapal).
- Semua nilai koordinat z diperlukan gambar bantuan yaitu *body plan* kapal dan bukaan kulit. Dari gambar bukaan kulit diperoleh posisi pelat datar pada kulit lambung kapal. Kemudian pelat datar tersebut digambarkan pada *body plan* sesuai dengan letaknya pada gambar bukaan kulit tersebut. Dari sini bisa dibaca besar nilai z.

##### ▪ Letak Horisontal

Pembacaan data desain pelat datar horisontal :

- Menentukan titik origin pada ujung pelat dengan posisi paling dekat dari *base line* dan paling dekat dari AP.
- Pembacaan nilai koordinat y sesuai letak titik sepanjang sumbu y (melintang kapal).
- Pembacaan nilai koordinat x sesuai letak titik sepanjang sumbu x (memanjang kapal).
- Nilai koordinat z semua titik adalah nol.

#### □ **Pelat Lengkung (Simetris dan Tidak Simetris)**

Pembacaan data desain pelat lengkung :

- Menentukan titik origin pada ujung pelat dengan posisi paling dekat dari *base line* dan paling dekat dari AP.
- Pembacaan nilai koordinat 3-D (x, y, z) berdasarkan letak masing-masing komponen pada lambung kapal (dari gambar bukaan kulit) yang digambarkan pada *body plan* kapal.

#### □ **Inner Part**

- Letak Melintang Kapal

Pembacaan data desain *inner part* :

- Menentukan titik origin pada salah satu ujung pelat.
- Pembacaan nilai koordinat x sesuai letak titik sepanjang sumbu y (melintang kapal).
- Nilai koordinat x semua titik adalah nol.
- Pembacaan nilai koordinat z sesuai letak titik sepanjang sumbu z (vertikal kapal).



- Letak Memanjang Kapal

Pembacaan data desain *inner part* :

- Menentukan titik origin pada salah satu ujung pelat.
- Nilai koordinat y semua titik adalah nol.
- Pembacaan nilai koordinat x sesuai letak titik sepanjang sumbu x (memanjang kapal).
- Pembacaan nilai koordinat z sesuai letak titik sepanjang sumbu z (vertikal kapal).

## **LAMPIRAN C**

### **MARKING TABLE**

#### **DATA-DATA TEKNIS**

- 1 buah marking off table tipe ZETTMESS

Terdiri dari 2 meja plat marking yang terbuat dari besi cor paduan baja, dengan keakurasian permukaan sesuai DIN 876/III serta dilengkapi dengan garis-garis grid yang berukuran 100 x 100 mm.

Ukuran total dari meja plat	9.000 x 3.400 mm
Ukuran tiap meja plat	4.500 x 3.400 mm
Tebal meja plat	350 mm

Tinggi area kerja (tinggi meja dari lantai) dapat antara 620 sampai 670 mm, dan untuk pengaturannya dilengkapi dengan 36 penyetelan height supports tipe ZETTMESS no 1 (270 - 320) terbuat dari besi cor dengan penyetelan sekrup M 30.

- 2 buah 3 (tiga) axis koordinat pengukuran dan marking tipe AKL 16 yang terdiri dari :
  - Guideways dari aluminium
  - Kompensasi dari crossarm (lengan marking)
  - Ketinggian (vertikal) axis Z 1.600 mm
  - Melintang axis Y 1.200 mm
  - Memanjang (longitudinal) axis X 8.500 mm
  - Penggerak kaki mesin pada plat/machine foot moving (axis X) dan



guidenya serta guide rail (penuntun rel) terbuat dari logam yang dikeraskan dan chromium plate. Terdiri dari 2 guide (penuntun) dengan panjang tiap guide 3 x 3.000 mm yang totalnya 9.000 mm.

- Peralatan dengan dial readout untuk axis X, Y, Z (TRAK 103)
- Scriber holder dan measuring probe
- Assesoris standart untuk mesin marking :
  - 1 scriber 220 mm
  - 1 scriber 270 mm
  - 1 scriber 320 mm
  - 1 circular marker dengan vernier dan interchangeable carbide insert
  - 1 carbide tipped spare
  - dial gauge dan holder

□ Assesoris :

- 1 Perpendicular granit made gauge 800 mm
- Spirit level khusus dengan keakurasian 0,02 mm per meter
- Hockgauge 750 mm
- Reference cube

## **INSTRUKSI PENGOPERASIAN UNTUK MARKING DAN MENGECEK MESIN**

□ **Pengenalan Mesin**

Telah diketahui bahwa, koordinat marking dan pengecekan mesin serta peralatan-peralatan tidak dapat dilepaskan dari pengoperasian-pengoperasian. Mesin telah disiapkan dan dilengkapi dengan peralatan-peralatan termasuk dalam mengatasi perubahan-perubahan yang terjadi sehingga dapat menjamin tingkat kepresisian dari koordinat marking dan pengecekan mesin.

Komponen-komponen mesin marking :

- *Master Column*

Pada sisi-sisi master column terdapat rubber/karet yang berfungsi untuk membersihkan sisi-sisi tersebut dari kotoran/debu dan juga terdapat scriber holder yang berfungsi untuk mengetahui nilai pada axis Z.

- *Cross Arm*

Pada sisi-sisi cross arm juga terdapat rubber/karet dan scriber holder seperti pada master column.

- *Machine foot moving plate*

Pada foot moving terdapat 3 pasang bearing, 2 pasang berada di belakang dan 1 pasang berada di depan. Fungsi bearing tersebut adalah untuk menjalankan foot moving. Dari 3 pasang bearing tersebut harus mempunyai tekanan yang sama dan untuk mengetahui tekanan itu dapat menggunakan Aluminium Foil.

#### Cara mengetahui tekanan bearing

Sebelum Aluminium Foil digunakan harus ditekan-tekan dahulu dengan tangan, setelah rata Aluminium Foil diletakkan pada sisi bearing kemudian menjalankan foot moving plate, maka setelah Aluminium Foil diambil akan terlihat bekas tekanan bearing tersebut. Hal ini dilakukan pada ketiga pasang bearing tersebut, jika salah satu bearing tekanannya berbeda maka langkah yang harus dikerjakan adalah membuka sekrup dengan kunci L dan menurunkan bearing dengan kunci khusus. (Dalam mengubah posisi bearing harus dikerjakan sedikit demi sedikit).



- *Rail dan guide*

Rail dipasang sepanjang marking table (panjang 9000 mm), tetapi panjang tiap rail adalah 3000 mm, jadi terdapat 3 rail pada satu sisi marking table. Untuk pemasangan rail pada marking plate menggunakan sekrup dan kunci L. Pada sisi samping rail terdapat track, dimana track tersebut dikaitkan dengan bearing yang fungsinya untuk menjalankan foot moving plate. Dan untuk mengetahui apakah antara bearing dan track tersebut longgar atau rapat yaitu dengan mendengar bunyi yang timbul dari jalannya foot moving plate.

Kedua komponen Master Column dan Cross Arm terbuat dari aluminium khusus yang dalam penyusunannya digabungkan dengan efek dari *Robust bearing* yang menjamin tingkat kepresisian penuh.

Berdirinya Mounting base yang terbuat dari besi cor dengan ukuran yang besar membuat mesin tetap berdiri pada posisi tegak.

- **Assesoris Marking Plate Dan Penggunaannya**

- **MARKING HEAD**

Marking Head Universal dapat dirangkai pada akhir Horizontal Arm. Dengan diputar  $90^\circ$ , head dapat digunakan untuk marking secara horizontal dan marking secara vertikal. Marking head terdiri dari 2 elemen yang dapat melakukan 3 pengoperasian marking untuk axis X, axis Y dan axis Z. Peralatan *centric needle reception* dapat berputar  $90^\circ$  untuk mengoperasikan marking pada axis horizontal dan vertikal. Peralatan *eccentric reception* memungkinkan dalam pengoperasian marking secara vertikal tanpa harus ada perputaran mesin.

### Kalibrasi Marking Head

Untuk marking head yang dikalibrasi adalah pada lengan (Arm) berikutnya adalah *Fixation head*, pengukuran paralel atau pengukuran angular yang diperoleh dengan mengoperasikan sekrup central pada permukaan dari Arm bagian atas atau bawah.

Dua sekrup yang saling berhadapan yang mengatur tekanan dilengkapi dengan *Periphery ring retainer* yang lebih kecil yang dioperasikan untuk mengkalibrasi cara putar  $90^{\circ}$ .

### Cara mengkalibrasi Marking Head

- Perpendicular dipasang pada marking head.
- Dial Gauge disentuhkan pada perpendicular, dan cross arm digerakkan ke atas/ke bawah, maka akan diketahui berapa besar kemiringan marking head tersebut, jika miring maka sekrup yang di atas harus distel.
- Untuk mengetahui posisi marking head apakah miring searah axis X, dan caranya juga sama seperti cara di atas tetapi cross arm dijalankan maju-mundur searah axis X dan sekrup yang distel adalah sekrup yang ada pada sisi samping.

#### ▪ CIRCULAR MARKING

Alat Circular Marking yang berdiameter 60 - 70 mm digunakan untuk marking, alat tersebut dikombinasi dengan Deflection block yang memungkinkan untuk memarking atau menandai setiap tempat. Sebagai penambahan, pada kenaikan tingkat  $15^{\circ}$  dilengkapi dengan lubang-lubang yang sesuai pada lingkaran skala.



Dua sekrup yang saling berhadapan mengatur tekanan dan dilengkapi dengan *Periphery ring retainer* yang lebih kecil, yang dioperasikan untuk mengkalibrasi cara putar  $90^\circ$  (sebelumnya tempat/posisi sekrup M3 harus dilepas atau dikencangkan).

Circular marking yang dikombinasi dengan Deflection block untuk diameternya dapat ditandai pada tiap alat.

#### Penggunaan dari alat Circular Marking

Berikutnya adalah pelurusan, memarking center dan menitik (membuat titik). Alat Circular marking yang dipasang, setelah radiusnya diset, titik pusat dari cross arm digerakkan maju menandai titik. Screen stop harus dilepas/dibebaskan dan clamping device dilepaskan sehingga measuring arm dapat berputar  $360^\circ$ .

Harap diperhatikan :

Kedua mesin (2 buah 3 axis koordinat pengukuran dan marking tipe AKL 16) dan cross arm harus tetap kaku/keras dan tegak selama pengoperasian marking.

#### ▪ DEFLECTION BLOCK $90^\circ$

Deflection block digunakan dengan menggabungkan marking head atau alat circular marking. Penggabungan ini memungkinkan pengoperasian marking dilakukan pada setiap pekerjaan. Sama seperti marking head atau circular marking, deflection block  $90^\circ$  dipasang dengan menempatkan block pada setpin bores dan mengeraskan (mengencangkan) sekrup socket hexagon.

## □ Kalibrasi

Pada prinsipnya setiap mesin dikalibrasi seluruhnya dan dicek awal pada waktu pengiriman pertama, sehingga dalam pemasangannya tidak menimbulkan masalah.

Jika pada mesin ditemukan ketidaktepatan yang mungkin disebabkan oleh faktor transportasi dan lain-lain, dimana hal ini dapat diatasi dengan pengukuran kalibrasi.

## ■ KALIBRASI MASTER COLUMN

Dimanapun prakteknya, Master column boleh diluruskan dengan alat dial gauge atau appropriate boring mill. Mesin berdiri dengan menyatukan 4 (empat) *enccentric bearing* pada sisi dinding dan 2 (dua) *enccentric bearing* yang lain pada sisi yang berlawanan. Pertama menggerakkan *enccentric bearing* dengan alat/kunci yang sudah termasuk dalam suplai barang, sekrup pengunci pada daerah atas harus dilepaskan. Pada kalibrasi Master column sekrup tersebut harus dikeraskan juga. Kemudian bar tekan digunakan untuk mengencangkan/mengatasi kondisi tersebut.

### Cara mengkalibrasi

- Dial gauge dipadang pada marking head.
- Stone angle diletakkan pada posisi berdiri.
  - Jarak diambil separuh dari nilai maksimum dari axis Z (800 mm).
  - Dial gauge ditarik ke bawah dan disentuhkan dengan stone angle kemudian diset nol.
  - Selanjutnya dial gauge dinaikkan sampai ujung stone angle maka akan





diketahui berapa nilai kemiringan dari master column tersebut.

- Jika master column miring ke arah push pul rod maka baut yang atas dilonggarkan kemudian baut yang bawah dikencangkan hingga mencapai kelurusan (dial gauge pada point nol).
- Jika master column miring menjauhi push pul rod maka baut yang dibawah dilonggarkan, kemudian baut yang di atas dikencangkan hingga mencapai kelurusan.
- Jika pada master column miring ke arah axis X maka berarti tekanan pada bearing (berada pada foot moving) berbeda antara bearing yang kiri dengan bearing yang kanan.
- Untuk kemiringan seperti ini menggunakan stone angle dan dial gauge untuk mengetahui berapa besar kemiringan dari master column tersebut. (Caranya juga sama seperti dia atas, tetapi letak stone angle diubah sesuai dengan kelurusan yang akan dikerjakan).
- Bila master column miring pada posisi 1, maka harus menurunkan bearing B yaitu membuka sekrup dengan kunci L dan memutar bearing agar turun, hingga dial gauge pada point nol. Langkah selanjutnya adalah mengerasi sekrup tersebut. Sebaliknya jika master column miring pada posisi 2 maka yang harus dilakukan adalah menurunkan bearing A.

#### ▪ KALIBRASI CROSS ARM

Cross arm tidak dapat dikalibrasi sampai *slide casing* pada mesin telah diluruskan dulu terlebih dahulu sebagai berikut :

Letakkan cross arm pada posisi tengah (pusat), luruskan permukaan cross arm

bagian bawah dengan alat dial gauge. 8 (delapan) pada permukaan bearing harus dikendori atau dikencangkan untuk memungkinkan gerakan yang diperlukan (sekrup A). Ke gerakan melingkar (angular) dicapai dengan mengendori atau mengencangkan sekrup pada permukaan bearing.

Roller tekan harus disusun menurut pertimbangan untuk setiap roller sehingga menjamin gerakan yang mudah dari *slide casing* dan rooler tidak harus mengencang. Selanjutnya adalah pelurusan slide casing dan cross arm juga diluruskan dengan cara sebagai berikut :

Tempatkan dial gauge pada bagian atas, cross arm digerakkan sesuai kepanjangan horisontal. Cross arm dikalibrasi dengan mengoperasikan sekrup tekan bagian atas. Pertama sekrup tekan harus dikendori, kemudian cap (tutup) plastik pada akhir dan awal harus digerakkan, sedangkan pul rod yang ada di dalam cross arm dikeraskan pelan-pelan dengan memutar baut M 12. Perbedaan ukuran distabilkan dengan mengeraskan sekrup tekan.

#### Cara mengkalibrasi

- Memasang dial gauge pada marking head, dimana ujung dial gauge harus menyentuh marking table.
- Cross arm dimajukan sampai maksimum, tetapi sekrup di bagian belakang cross arm harus terlihat sebab sekrup tersebut digunakan untuk menyetel cross arm.
- Setelah ujung dial gauge menyentuh marking table, maka langkah selanjutnya adalah memutar dial gauge pada point nol.
- Kemudian menarik cross arm sampai pada ujungnya, maka akan diketahui berapa besar kemiringan dari cross arm tersebut.



- Jika cross arm miring ke bawah harus melonggarkan sekrup yang ada di belakang cross arm hingga mencapai kelurusan (dial gauge pada point nol). Sebaliknya jika cross arm miring ke atas, maka sekrup harus dikerasi hingga mencapai kelurusan.

## **PEMELIHARAAN (MAINTENANCE) MESIN**

Pengoperasian dengan hati-hati akan menjamin umur dari mesin agar bertahan lama. Debu dan kotoran logam mempunyai pengaruh yang kurang baik terhadap pengoperasian. Permukaan dari mesin harus dibersihkan setiap hari dengan menggunakan campuran *resin-free machine oil* (2 bagian) dan petroleum (8 bagian). Pembersihan seluruhnya dari guide tracks dengan kain lembut (majun) yang dapat cepat menyerap. Dan dengan campuran di atas dapat mencegah debu dari penetrasi bearing dan bermacam-macam kotoran pada logam guide tracks. Mesin dilengkapi dengan Trans-A Dial atau sistem pengukuran Trak 103 dan harus dicek perbedaan langkahnya. Alat pengupas (stripper) yang dilengkapi dengan sensor harus diberi oil dengan beberapa tetes *resin-free machine oil* pada setiap satu minggu sekali secara berkala. Pengecekan bearing dilakukan setiap 3 atau 4 tahun sekali.

## **DIAL READOUT TRAK 103**

Dial readout trak 103 menggunakan 2 koordinat yaitu secara incremental dan secara absolut, dengan dapat memilih pemisahan dan output pada setiap axis. Dial readout trak 103 dilengkapi dengan kapasitas *make-learn* dimana dapat digunakan untuk menyimpan program secara otomatis.

Bagian-bagian dari Dial readout trak 103 adalah :

- Keyboard
- Input pengoperasian trak 103 melalui tombol karet keyboard yang ada pada panel depan (lihat gambar 8 pada lampiran 5).
- X, Y, Z : pilihan axis untuk perintah selanjutnya.
- \* : memulai perintah sistem konfigurasi.
- PRESET : memuat data dan memasukkan mode "LEARN", "RUN" atau "DRO".
- INC/ABS : Mengubah satu atau semua axis dari incremental ke absolut atau dari absolut ke incremental.
- RSTR (stands for restore) : digunakan untuk memulai sistem atau memasukkan data.
- M : tombol dari mode "DRO" ke "RUN" ke "LEARN".
- TOOL SET : memasukkan offset tool.
- TOOL : memilih nomor tool.
- +/- : mengganti data input dari "+" ke "-" atau sebaliknya.
- 0-9,. : data-data yang dimasukkan.
- Back Panel (Panel belakang)

Pada back panel (panel belakang) trak 103 berisi switch ON/OFF, power input connector, sensor connector dan test port connector.



## LAMPIRAN D

### LISTING PROGRAM MATCHING PROCESS

#### 1. VISUAL BASIC

##### MAIN.Frm (Form 1)

(Declaration)

```
Dim Hasil, n, a, i, j As Integer
Dim ja(1000) As Single
Dim jb(1000) As Single
Dim jc(1000) As Single
Dim jd(1000) As Single
Dim je(1000) As Single
Dim jf(1000) As Single
Dim c3(1000) As Single
Dim Awal(1000, 1000) As Single
Dim y11(1000, 1000) As Single
Dim y12(1000, 1000) As Single
Dim A1, A2, A3, A4 As String
Dim i1, i2 As String
Dim batal As String
Dim batal1 As String
Dim nomorfile
Dim namafile As String
Dim X As String
Dim tl1, tl2 As String
Dim P, pt, st, FL, vp, c, apl, SelAwal
```

Private Sub Form\_Load()

```
Screen.MousePointer = 11
If FL <> 1 Then
Form0.Show
pt = 6
st = Timer
Do While Timer < st + pt
DoEvents
Loop
Unload Form0
Else
End If
```

```
ChDir App.Path
ChDrive App.Path
```



```
Top = Screen.Height / 2 - Height / 2
Left = Screen.Width / 2 - Width / 2
```

```
If namafile = "" Then
    namafile = "unnamed"
End If
```

```
If Form1.Caption = "" Then
    Form1.Caption = "MATCHING PROCESS - [" & namafile & "]"
End If
```

```
Command2.Enabled = False
Command3.Enabled = False
Command4.Enabled = False
Command5.Enabled = False
Command6.Enabled = False
Command7.Enabled = False
```

'Tabel

```
Grid1.Row = 1
Grid1.Cols = 4
Grid1.FixedCols = 1
Grid2.Row = 1
Grid2.Cols = 4
Grid2.FixedCols = 1
```

'Tabel 1

```
Grid1.Row = 0
Grid1.Col = 0
Grid1.FixedAlignment(0) = 2: Grid1.Text = "(M,N)"
Grid1.Col = 1
Grid1.FixedAlignment(1) = 2: Grid1.Text = "X"
Grid1.Col = 2
Grid1.FixedAlignment(2) = 2: Grid1.Text = "Y"
Grid1.Col = 3
Grid1.FixedAlignment(3) = 2: Grid1.Text = "Z"
Grid1.Col = 3
Grid1.Row = 1
Grid1.Text = ""
```

```
For i = 1 To 3
    Grid1.ColWidth(i) = 1000
Next i
```

```
Grid1.ColWidth(0) = 600
```

'Tabel 2

```
Grid2.Row = 0
Grid2.Col = 0
```



```
Grid2.FixedAlignment(0) = 2: Grid2.Text = "(M,N)"
Grid2.Col = 1
Grid2.FixedAlignment(1) = 2: Grid2.Text = "X"
Grid2.Col = 2
Grid2.FixedAlignment(2) = 2: Grid2.Text = "Y"
Grid2.Col = 3
Grid2.FixedAlignment(3) = 2: Grid2.Text = "Z"
Grid2.Col = 3
Grid2.Row = 1
Grid2.Text = ""
```

```
For i = 1 To 3
    Grid2.ColWidth(i) = 1000
Next i
```

```
Grid2.ColWidth(0) = 600
```

```
Screen.MousePointer = 0
```

```
End Sub
```

```
Private Sub btabel()
```

```
Text(0).Text = Text4(0).Text * Text1(1).Text
Grid1.Rows = Val(Text(0).Text) + 1
Grid2.Rows = Val(Text(0).Text) + 1
```

```
For i = 1 To 3
    For j = 1 To Val(Text(0).Text)
        Grid1.Row = j
        Grid1.Col = i: Grid1.ColAlignment(i) = 1
        Grid2.Row = j
        Grid2.Col = i: Grid2.ColAlignment(i) = 1
    Next
Next
```

```
n = 0
For i = 1 To Val(Text4(0).Text)
    For j = 1 To Val(Text1(1).Text)
        Grid1.Col = 0: n = n + 1: Grid1.Row = n: Grid1.FixedAlignment(0) = 2
        Grid1.Text = "(" & i - 1 & "," & j - 1 & ")"
    Next
Next
```

```
n = 0
For i = 1 To Val(Text4(0).Text)
    For j = 1 To Val(Text1(1).Text)
        Grid2.Col = 0: n = n + 1: Grid2.Row = n: Grid2.FixedAlignment(0) = 2
        Grid2.Text = "(" & i - 1 & "," & j - 1 & ")"
    Next
Next
```

Next

Next

End Sub

Private Sub bukafile()

FL = 1

dlgbukasimpan.DialogTitle = "Buka File"

nomorfile = FreeFile

dlgbukasimpan.InitDir = ""

dlgbukasimpan.Filter = "File Data(\*.dat)|\*.dat"

dlgbukasimpan.FilterIndex = 2

dlgbukasimpan.Action = 1

namafile = dlgbukasimpan.filename

If namafile = "" Then

Exit Sub

End If

Form1.Caption = "MATCHING PROCESS - [" & namafile & "]"

Command2.Enabled = True

Command3.Enabled = True

Command4.Enabled = False

Command5.Enabled = False

Command6.Enabled = False

Command7.Enabled = False

Open namafile For Input As nomorfile

Do Until EOF(nomorfile)

Input #nomorfile, A1, A2

Text(0).Text = A1: Text1(1).Text = A2

Input #nomorfile, A3, A4

Text4(0).Text = A3: text5.Text = Format(A4, "0.000")

btabel

For i = 1 To 3

Grid1.Col = i

For j = 1 To A1

Input #nomorfile, isi1

Grid1.Row = j

Grid1.Text = Format(isi1, "0.000")

Next

Next

For i = 1 To 3

Grid2.Col = i

For j = 1 To A1



```
Input #nomorfile, isi2
Grid2.Row = j
Grid2.Text = Format(isi2, "0.000")
Next
Next
Loop
```

```
Close #nomorfile
Text4(0).SetFocus
btabel
```

End Sub

Private Sub filetxt1()

```
X = "X"
nomorfile = FreeFile
Open "matching1.txt" For Output As nomorfile
Write #nomorfile, Val(Text4(0).Text)
Write #nomorfile, Val(Text1(1).Text)
```

```
M = 0
For i = 1 To 3
For j = 1 To Text(0).Text
Grid1.Col = i: Grid1.Row = j
mmax = Val(Grid1.Text)
If Val(Grid1.Text) < 0 Then
mmax = Val(Grid1.Text) * (-1)
End If
If M < mmax Then
M = mmax
Elseif M = mmax Then
M = mmax
Elseif M > mmax Then
M = M
End If
Next
Next
```

```
n = 0
For i = 1 To 3
For j = 1 To Text(0).Text
Grid2.Col = i: Grid2.Row = j
nmax = Val(Grid2.Text)
If Val(Grid2.Text) < 0 Then
nmax = Val(Grid2.Text) * (-1)
End If
If n < nmax Then
n = nmax
```

```
Elseif n = nmax Then
    n = nmax
Elseif n > nmax Then
    n = n
End If
Next
Next
```

```
If M > n Then
    M = M
Elseif M = n Then
    M = n
Elseif M < n Then
    M = n
End If
```

```
z = 0
For i = 1 To Text(0).Text
    Grid1.Col = 3: Grid1.Row = i
    zmax = Val(Grid1.Text)
    If z < zmax Then
        z = zmax
    Elseif z = zmax Then
        z = zmax
    Elseif z > zmax Then
        z = z
    End If
Next
```

```
Ma = M + (M / 10)
zm = z + Ma / 2
```

```
If M > zm Then
    M = M
Elseif M = zm Then
    M = zm
Elseif M < zm Then
    M = zm
End If
```

```
M0 = M + (M / 10)
Write #nomorfile, M0, 0, 0
Write #nomorfile, 0, M0, 0
Write #nomorfile, 0, 0, M0
```

```
M1 = M0 + (M / 20)
Write #nomorfile, M1, 0, 0
Write #nomorfile, 0, M1, 0
Write #nomorfile, 0, 0, M1
```



```

txt = M / 20
Write #nomorfile, txt

For e = 1 To Text(0).Text
    Grid1.Row = e
    For h = 1 To 3
        If h = 1 Then
            Grid1.Col = 1: ja(e) = Val(Grid1.Text)
        Elseif h = 2 Then
            Grid1.Col = 2: jb(e) = Val(Grid1.Text)
        Elseif h = 3 Then
            Grid1.Col = 3: jc(e) = Val(Grid1.Text)
        End If
    Next h
    Write #nomorfile, ja(e), jb(e), jc(e)
Next e
Print #nomorfile, X

```

```

For e = 1 To Text(0).Text
    Grid2.Row = e
    For h = 1 To 3
        If h = 1 Then
            Grid2.Col = 1: jd(e) = Val(Grid2.Text)
        Elseif h = 2 Then
            Grid2.Col = 2: je(e) = Val(Grid2.Text)
        Elseif h = 3 Then
            Grid2.Col = 3: jf(e) = Val(Grid2.Text)
        End If
    Next h
    Write #nomorfile, jd(e), je(e), jf(e)
Next e
Print #nomorfile, X
Close #nomorfile

```

End Sub

Private Sub gambar()

```

Image1.Picture = LoadPicture("")
tl1 = "bc1": tl2 = "bc2"
If apl <> 1 Then
    Shell "c:/progra~1/r13/win/acad.exe", 4
    Screen.MousePointer = 11
    pt = 16 ' Set duration.
    st = Timer ' Set start time.
    Do While Timer < st + pt
    DoEvents
    Loop

```

```
End If
apl = 1
If apl = 1 Then
AppActivate "AutoCAD - [unnamed]"
SendKeys tl1 & "{ENTER}", True
SendKeys tl2 & "{ENTER}", True
End If
apl = 1
```

```
pt = 6
st = Timer
Do While Timer < st + pt
DoEvents
Loop
Image1.Picture = LoadPicture("NE2.bmp")
```

```
pt = 6
st = Timer
Do While Timer < st + pt
DoEvents
Loop
Image1.Picture = LoadPicture("NE1.bmp")
```

```
Screen.MousePointer = 0
```

```
End Sub
```

```
Private Sub simpanfile()
```

```
P = 2
If Text1(1) = "" Or Text4(0) = "" Or Text(0) = "" Then
MsgBox "Data tidak lengkap !", 0 + 48, "Pesan"
Text4(0).SetFocus
Screen.MousePointer = 0
Exit Sub
End If
```

```
dlgbukasimpan.DialogTitle = "Simpan File"
nomorfile = FreeFile
dlgbukasimpan.InitDir = ""
dlgbukasimpan.Filter = "Semua file (*.*)|*..*|file Data (*.dat)|*.dat"
dlgbukasimpan.FilterIndex = 2
dlgbukasimpan.Action = 2
namafile = dlgbukasimpan.filename
```

```
If Dir(namafile) <> "" Then
pilihan = MsgBox("File sudah ada ! Simpan di tempat yang sama ?", 3 + 48,
"Pesan")
If pilihan = 6 Then
```



```

    Screen.MousePointer = 0
ElseIf pilihan = 2 Then
    Screen.MousePointer = 0
Exit Sub
Else
    dlgbukasimpan.DialogTitle = "Simpan File"
    nomorfile = FreeFile
    dlgbukasimpan.InitDir = ""
    dlgbukasimpan.Filter = "Semua file (*.*)|*. *|file Data (*.dat)|*.dat"
    dlgbukasimpan.FilterIndex = 2
    dlgbukasimpan.Action = 2
    namafile = dlgbukasimpan.filename
End If
End If

```

```

btabel
Open namafile For Output As nomorfile
Write #nomorfile, Val(Text(0).Text), Val(Text1(1).Text)
Write #nomorfile, Val(Text4(0).Text), Val(text5.Text)
For ii = 1 To 3
    For jj = 1 To Val(Text(0).Text)
        Grid1.Col = ii: Grid1.Row = jj
        If Grid1.Text = "" Then
            isi1 = 0
        Else
            isi1 = Val(Grid1.Text)
        End If
        Write #nomorfile, isi1
    Next
Next

```

```

For ii = 1 To 3
    For jj = 1 To Val(Text(0).Text)
        Grid2.Col = ii: Grid2.Row = jj
        If Grid2.Text = "" Then
            isi2 = 0
        Else
            isi2 = Val(Grid2.Text)
        End If
        Write #nomorfile, isi2
    Next
Next
Close #nomorfile
Form1.Caption = "MATCHING PROCESS - [" & namafile & "]"

Screen.MousePointer = 0

```

```

End Sub

```

```
Private Sub Command1_Click()
```

```
    If IsNumeric(Text4(0).Text) = False Then  
        MsgBox "Isi jumlah baris (M) !", 0 + 48, "Pesan"  
        Text4(0).SetFocus: Text4(0).Text = ""  
    Exit Sub  
End If
```

```
    If Text4(0) = 1 Then  
        MsgBox "Jumlah baris (M) > 1 !", 0 + 48, "Pesan"  
        Text4(0).SetFocus: Text4(0) = ""  
    Exit Sub  
End If
```

```
    If IsNumeric(Text1(1).Text) = False Then  
        MsgBox "Isi jumlah kolom (N) !", 0 + 48, "Pesan"  
        Text1(1).SetFocus: Text1(1).Text = ""  
    Exit Sub  
End If
```

```
    If Text1(1) = 1 Then  
        MsgBox "Jumlah kolom (N) > 1 !", 0 + 48, "Pesan"  
        Text1(1).SetFocus: Text1(1) = ""  
    Exit Sub  
End If
```

```
    If IsNumeric(Text4(0).Text) = True And IsNumeric(Text1(1).Text) = True Then  
        Screen.MousePointer = 11  
        btabel  
        Grid1.Row = 1  
        Grid1.Col = 2  
        Screen.MousePointer = 0  
        Command2.Enabled = True  
        Command3.Enabled = True  
    Exit Sub  
End If
```

```
End Sub
```

```
Private Sub Command1_Click()
```

```
    If IsNumeric(Text4(0).Text) = False Then  
        MsgBox "Isi jumlah baris (M) !", 0 + 48, "Pesan"  
        Text4(0).SetFocus: Text4(0).Text = ""  
    Exit Sub  
End If
```

```
    If Text4(0) = 1 Then  
        MsgBox "Jumlah baris (M) > 1 !", 0 + 48, "Pesan"
```

```
Text4(0).SetFocus: Text4(0) = ""  
Exit Sub  
End If
```

```
If IsNumeric(Text1(1).Text) = False Then  
    MsgBox "Isi jumlah kolom (N) !", 0 + 48, "Pesan"  
    Text1(1).SetFocus: Text1(1).Text = ""  
Exit Sub  
End If
```

```
If Text1(1) = 1 Then  
    MsgBox "Jumlah kolom (N) > 1 !", 0 + 48, "Pesan"  
    Text1(1).SetFocus: Text1(1) = ""  
Exit Sub  
End If
```

```
If IsNumeric(Text4(0).Text) = True And IsNumeric(Text1(1).Text) = True Then  
    Screen.MousePointer = 11  
    btablel  
    Grid1.Row = 1  
    Grid1.Col = 2  
    Screen.MousePointer = 0  
    Command2.Enabled = True  
    Command3.Enabled = True  
Exit Sub  
End If
```

```
End Sub
```

```
Private Sub Command2_KeyDown(KeyCode As Integer, Shift As Integer)  
  
    Select Case KeyCode  
    Case &H27 'KeyRight  
        Command3.SetFocus  
    End Select
```

```
End Sub
```

```
Private Sub Command3_Click()
```

```
    If IsNumeric(text5.Text) = False Then  
        MsgBox "Isi toleransi !", 0 + 48, "Pesan"  
        text5.SetFocus: text5.Text = ""  
    Exit Sub  
End If
```

```
    For i = 1 To 3  
        For j = 1 To Val(Text(0).Text)  
            Grid2.Col = i: Grid2.Row = j
```



```

Grid1.Col = i: Grid1.Row = j
y11(i, j) = Val(Grid2.Text): y12(i, j) = Val(Grid1.Text)
Awal(i, j) = y11(i, j) - y12(i, j)
If Awal(i, j) < 0 Then
    SelAwal = Awal(i, j) * (-1)
Elseif Awal(i, j) >= 0 Then
    SelAwal = Awal(i, j)
End If
Next
Next

```

```

If SelAwal > Val(text5.Text) Then
    MsgBox "Data salah !", 0 + 48, "Pesan"
Exit Sub
Else
    Screen.MousePointer = 11
btabel
Form2.Show
Screen.MousePointer = 0
End If

```

End Sub

Private Sub Command3\_KeyDown(KeyCode As Integer, Shift As Integer)

```

Select Case KeyCode
Case &H25 'KeyLeft
Command2.SetFocus
End Select

```

```

Select Case KeyCode
Case &H26 'KeyUp
Command2.SetFocus
End Select

```

End Sub

Private Sub Command4\_Click()

```

Screen.MousePointer = 11
Image1.Picture = LoadPicture("")
Command4.Enabled = False
Command5.Enabled = True
Command6.Enabled = True
Command7.Enabled = True
Image1.Picture = LoadPicture("NE2.bmp")
pt = 6
st = Timer
Do While Timer < st + pt

```

```
DoEvents
Loop
Image1.Picture = LoadPicture("NE1.bmp")
Screen.MousePointer = 0
```

End Sub

```
Private Sub Command5_Click()
```

```
Screen.MousePointer = 11
Image1.Picture = LoadPicture("")
Command5.Enabled = False
Command4.Enabled = True
Command6.Enabled = True
Command7.Enabled = True
Image1.Picture = LoadPicture("NW2.bmp")
pt = 6
st = Timer
Do While Timer < st + pt
DoEvents
Loop
Image1.Picture = LoadPicture("NW1.bmp")
Command5.Enabled = False
Screen.MousePointer = 0
```

End Sub

```
Private Sub Command6_Click()
```

```
Screen.MousePointer = 11
Image1.Picture = LoadPicture("")
Command6.Enabled = False
Command5.Enabled = True
Command4.Enabled = True
Command7.Enabled = True
Image1.Picture = LoadPicture("SW2.bmp")
pt = 6
st = Timer
Do While Timer < st + pt
DoEvents
Loop
Image1.Picture = LoadPicture("SW1.bmp")
Screen.MousePointer = 0
```

End Sub

```
Private Sub Command7_Click()
```

```
    Screen.MousePointer = 11
    Image1.Picture = LoadPicture("")
    Command7.Enabled = False
    Command5.Enabled = True
    Command6.Enabled = True
    Command4.Enabled = True
    Image1.Picture = LoadPicture("SE2.bmp")
    pt = 6
    st = Timer
    Do While Timer < st + pt
    DoEvents
    Loop
    Image1.Picture = LoadPicture("SE1.bmp")
    Screen.MousePointer = 0
```

```
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)
```

```
    Screen.MousePointer = 11
    If P = 1 Then
        pilihan = MsgBox("Anda ingin menyimpan file " & namafile & " ini ?", 3 + 48,
        "Pesan")
        If pilihan = 6 Then
            simpanfile
            Screen.MousePointer = 0
        ElseIf pilihan = 2 Then
            Cancel = True
            Screen.MousePointer = 0
        Exit Sub
        Else
            Screen.MousePointer = 0
        End If
    End If

    If P <> 1 Then
        Screen.MousePointer = 0
    End If

    If apl = 1 Then
        AppActivate "AutoCAD - [unnamed]"
        SendKeys "x" & "{ENTER}", True
    End If
```

```
Unload Me
```

```
Unload Form0: Unload Form2
```

```
Unload Form3a: Unload Form3b: Unload Form3c
```



Unload Form4a: Unload Form4b: Unload Form4c  
Unload Form5

End Sub

Private Sub Grid1\_DblClick()

Dim KeyAscii As Integer

Text2.SetFocus  
SendKeys "{END}"  
KeyAscii = 13

End Sub

Private Sub Grid1\_KeyDown(KeyCode As Integer, Shift As Integer)

Select Case KeyCode  
Case &H71 'F2  
Text2.SetFocus  
SendKeys "{END}"  
End Select

End Sub

Private Sub Grid1\_KeyPress(KeyAscii As Integer)

Text2.SetFocus  
Text2 = ""  
SendKeys Chr\$(KeyAscii)

End Sub

Private Sub Grid1\_SelChange()

Text2.Text = Format(Grid1.Text, "0.000")

End Sub

Private Sub Grid2\_DblClick()

Dim KeyAscii As Integer

Text3.SetFocus  
SendKeys "{END}"  
KeyAscii = 13

End Sub

```
Private Sub Grid2_KeyDown(KeyCode As Integer, Shift As Integer)
```

```
    Select Case KeyCode  
        Case &H71 'F2  
            Text3.SetFocus  
            SendKeys "{END}"  
    End Select
```

```
End Sub
```

```
Private Sub Grid2_KeyPress(KeyAscii As Integer)
```

```
    Text3.SetFocus  
    Text3 = ""  
    SendKeys Chr$(KeyAscii)
```

```
End Sub
```

```
Private Sub Grid2_SelChange()
```

```
    Text3.Text = Format(Grid2.Text, "0.000")
```

```
End Sub
```

```
Private Sub mnubukafile_Click()
```

```
    Screen.MousePointer = 11
```

```
    If P = 1 Then
```

```
        pilihan = MsgBox("Anda ingin menyimpan file " & namafile & " ini ?", 3 + 48,  
"Pesan")
```

```
        If pilihan = 6 Then
```

```
            simpanfile
```

```
            Screen.MousePointer = 0
```

```
        ElseIf pilihan = 2 Then
```

```
            Screen.MousePointer = 0
```

```
        Exit Sub
```

```
        Else
```

```
            Screen.MousePointer = 0
```

```
        End If
```

```
    End If
```

```
    If P <> 1 Then
```

```
        Screen.MousePointer = 0
```

```
    End If
```

```
    bukafile
```

```
    P = 2
```

```
End Sub
```

```
Private Sub mnufilebaru_Click()
```

```
    Screen.MousePointer = 11
```

```
    FL = 1
```

```
    If P = 1 Then
```

```
        pilihan = MsgBox("Anda ingin menyimpan file " & namafile & " ini ?", 3 + 48,  
"Pesan")
```

```
        If pilihan = 6 Then
```

```
            simpanfile
```

```
            If Text1(1) = "" Or Text4(0) = "" Or Text(0) = "" Then
```

```
                Exit Sub
```

```
            End If
```

```
            Screen.MousePointer = 0
```

```
        ElseIf pilihan = 2 Then
```

```
            Screen.MousePointer = 0
```

```
        Exit Sub
```

```
        Else
```

```
            Screen.MousePointer = 0
```

```
        End If
```

```
    End If
```

```
    If P <> 1 Then
```

```
        Screen.MousePointer = 0
```

```
    End If
```

```
    P = 2
```

```
    Unload Form1
```

```
    Load Form1: Form1.Show
```

```
End Sub
```

```
Private Sub mnukeluar_Click()
```

```
    Screen.MousePointer = 11
```

```
    If P = 1 Then
```

```
        pilihan = MsgBox("Anda ingin menyimpan file " & namafile & " ini ?", 3 + 48,  
"Pesan")
```

```
        If pilihan = 6 Then
```

```
            simpanfile
```

```
            Screen.MousePointer = 0
```

```
        ElseIf pilihan = 2 Then
```

```
            Screen.MousePointer = 0
```

```
        Exit Sub
```

```
        Else
```

```
            Screen.MousePointer = 0
```

```
        End If
```

```
    End If
```

```
    If P <> 1 Then
```



```

        Screen.MousePointer = 0
    End If
    Unload Me

End Sub

Private Sub mnuketerangan_Click()

    Screen.MousePointer = 11
    Form5.Show
    Screen.MousePointer = 0

End Sub

Private Sub mnusimpanfile_Click()

    Screen.MousePointer = 11
    simpanfile
    Screen.MousePointer = 0

End Sub

Private Sub Text1_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)

    P = 1
    Select Case KeyCode
        Case &H41 To &H5A 'Huruf
            MsgBox "Masukkan bilangan !", 0 + 48, "Pesan"
            Text1(1).SetFocus: Text1(1) = ""
        Exit Sub
    End Select

    Select Case KeyCode
        Case &HD 'Enter
            text5.SetFocus
        End Select

    Select Case KeyCode
        Case &H26 'KeyUp
            Text4(0).SetFocus
        End Select

    Select Case KeyCode
        Case &H28 'KeyDown
            text5.SetFocus
        End Select

End Sub

```

```
Private Sub Text2_Change()
```

```
    P = 1
```

```
    Grid1.Text = Format(Text2.Text, "0.000")
```

```
End Sub
```

```
Private Sub Text2_Click()
```

```
    Dim KeyAscii As Integer
```

```
    Text2.SetFocus
```

```
    SendKeys "{END}"
```

```
    KeyAscii = 0
```

```
End Sub
```

```
Private Sub Text2_GotFocus()
```

```
    batal1 = Text2.Text
```

```
End Sub
```

```
Private Sub Text2_KeyDown(KeyCode As Integer, Shift As Integer)
```

```
    Select Case KeyCode
```

```
        Case &H41 To &H5A 'Huruf
```

```
            MsgBox "Masukkan bilangan !", 0 + 48, "Pesan"
```

```
            Grid1.SetFocus: Grid1 = ""
```

```
    End Select
```

```
    Select Case KeyCode
```

```
        Case &H1B 'Esc
```

```
            Text2.Text = batal1
```

```
            Grid1.SetFocus
```

```
    End Select
```

```
End Sub
```

```
Private Sub Text2_KeyPress(KeyAscii As Integer)
```

```
    Select Case KeyAscii
```

```
        Case &HD 'Enter
```

```
            Grid1.SetFocus
```

```
            KeyAscii = 0
```

```
            SendKeys "{DOWN}"
```

```
    End Select
```

```
End Sub
```

```
Private Sub Text3_Change()
```

```
    P = 1
```

```
    Grid2.Text = Format(Text3.Text, "0.000")
```

```
End Sub
```

```
Private Sub Text3_Click()
```

```
    Dim KeyAscii As Integer
```

```
    Text3.SetFocus
```

```
    SendKeys "{END}"
```

```
    KeyAscii = 0
```

```
End Sub
```

```
Private Sub Text3_GotFocus()
```

```
    Batal2 = Text3.Text
```

```
End Sub
```

```
Private Sub Text3_KeyDown(KeyCode As Integer, Shift As Integer)
```

```
    Select Case KeyCode
```

```
        Case &H41 To &H5A 'Huruf
```

```
        MsgBox "Masukkan bilangan !", 0 + 48, "Pesan"
```

```
        Grid2.SetFocus: Grid2 = ""
```

```
    End Select
```

```
    Select Case KeyCode
```

```
        Case &H1B 'Esc
```

```
        Text3.Text = Batal2
```

```
        Grid2.SetFocus
```

```
    End Select
```

```
End Sub
```

```
Private Sub Text3_KeyPress(KeyAscii As Integer)
```

```
    Select Case KeyAscii
```

```
        Case &HD 'Enter
```

```
        Grid2.SetFocus
```

```
        KeyAscii = 0
```

```
        SendKeys "{down}"
```

```
    End Select
```

```
End Sub
```



```
Private Sub Text4_KeyDown(Index As Integer, KeyCode As Integer, Shift As Integer)
```

```
    P = 1
    Select Case KeyCode
    Case &H41 To &H5A 'Huruf
    MsgBox "Masukkan bilangan !", 0 + 48, "Pesan"
    Text4(0).SetFocus: Text4(0) = ""
    Exit Sub
    End Select
```

```
    Select Case KeyCode
    Case &HD 'Enter
    Text1(1).SetFocus
    End Select
```

```
    Select Case KeyCode
    Case &H28 'KeyDown
    Text1(1).SetFocus
    End Select
```

```
End Sub
```

```
Private Sub Text5_KeyDown(KeyCode As Integer, Shift As Integer)
```

```
    P = 1
    Select Case KeyCode
    Case &H41 To &H5A 'Huruf
    MsgBox "Masukkan Bilangan !", 0 + 48, "Pesan"
    text5.SetFocus: text5 = ""
    End Select
```

```
    Select Case KeyCode
    Case &HD 'Enter
    Command1.SetFocus
    End Select
```

```
    Select Case KeyCode
    Case &H26 'KeyUp
    Text1(1).SetFocus
    End Select
```

```
    Select Case KeyCode
    Case &H28 'KeyDown
    Command1.SetFocus
    End Select
```

```
End Sub
```

```
Private Sub filetxt2()
```

```
    X = "X"  
    nomorfile = FreeFile  
    Open "matching2.txt" For Output As nomorfile  
    Write #nomorfile, Val(Text4(0).Text)  
    Write #nomorfile, Val(Text1(1).Text)
```

```
    M = 0  
    For i = 1 To 3  
        For j = 1 To Text(0).Text  
            Grid1.Col = i: Grid1.Row = j  
            mmax = Val(Grid1.Text)  
            If Val(Grid1.Text) < 0 Then  
                mmax = Val(Grid1.Text) * (-1)  
            End If  
            If M < mmax Then  
                M = mmax  
            ElseIf M = mmax Then  
                M = mmax  
            ElseIf M > mmax Then  
                M = M  
            End If  
        Next  
    Next
```

```
    n = 0  
    For i = 1 To 3  
        For j = 1 To Text(0).Text  
            Grid2.Col = i: Grid2.Row = j  
            nmax = Val(Grid2.Text)  
            If Val(Grid2.Text) < 0 Then  
                nmax = Val(Grid2.Text) * (-1)  
            End If  
            If n < nmax Then  
                n = nmax  
            ElseIf n = nmax Then  
                n = nmax  
            ElseIf n > nmax Then  
                n = n  
            End If  
        Next  
    Next
```

```
    If M > n Then  
        M = M  
    ElseIf M = n Then  
        M = n  
    ElseIf M < n Then
```

```
M = n
End If
```

```
z = 0
For i = 1 To Text(0).Text
    Grid1.Col = 3: Grid1.Row = i
    zmax = Val(Grid1.Text)
    If z < zmax Then
        z = zmax
    ElseIf z = zmax Then
        z = zmax
    ElseIf z > zmax Then
        z = z
    End If
Next
```

```
Ma = M + (M / 10)
zm = z + Ma / 2
```

```
If M > zm Then
    M = M
Elseif M = zm Then
    M = zm
Elseif M < zm Then
    M = zm
End If
```

```
M0 = M + (M / 10)
Write #nomorfile, M0, 0, 0
Write #nomorfile, 0, M0, 0
Write #nomorfile, 0, 0, M0
```

```
M1 = M0 + (M / 20)
Write #nomorfile, M1, 0, 0
Write #nomorfile, 0, M1, 0
Write #nomorfile, 0, 0, M1
```

```
txt = M / 20
Write #nomorfile, txt
```

```
For e = 1 To Text(0).Text
    Grid1.Row = e
    For h = 1 To 3
        If h = 1 Then
            Grid1.Col = 1: ja(e) = Val(Grid1.Text)
        ElseIf h = 2 Then
            Grid1.Col = 2: jb(e) = Val(Grid1.Text)
        ElseIf h = 3 Then
            Grid1.Col = 3: jc(e) = Val(Grid1.Text)
        End If
    Next h
Next e
```



```

    End If
Next h
    Write #nomorfile, ja(e), jb(e), jc(e)
Next e
Print #nomorfile, X

For e = 1 To Text(0).Text
    Grid2.Row = e
    For h = 1 To 3
        If h = 1 Then
            Grid2.Col = 1: jd(e) = Val(Grid2.Text)
        ElseIf h = 2 Then
            Grid2.Col = 2: je(e) = Val(Grid2.Text)
        ElseIf h = 3 Then
            For j = 1 To Text(0).Text
                Grid1.Col = 3
                Grid2.Col = 3
                c3(j) = z - Val(Grid1.Text)
                jf(e) = Val(Grid2.Text) + c3(j) + M0 / 2
            Next
        End If
    Next h
    Write #nomorfile, jd(e), je(e), jf(e)
Next e
Print #nomorfile, X
Close #nomorfile

```

End Sub

## HASIL.Frm (Form 2)

(Declaration)

```

Dim i, j, k As Integer
Dim y11(1000, 1000) As Single
Dim y12(1000, 1000) As Single
Dim IsiX(1000), IsiY(1000), IsiZ(1000) As Single
Dim SelAwal(1000, 1000) As Single
Dim Awal(1000, 1000) As Single
Dim HasilX, HasilY, HasilZ
Dim minX, minY, minZ

```

Private Sub Form\_Load()

'Perhitungan

```

For i = 1 To 3
    For j = 1 To Val(Form1!Text(0).Text)
        Form1!Grid2.Col = i: Form1!Grid2.Row = j
    
```

```

Form1!Grid1.Col = i: Form1!Grid1.Row = j
y11(i, j) = Val(Form1!Grid2.Text): y12(i, j) = Val(Form1!Grid1.Text)
Awal(i, j) = y11(i, j) - y12(i, j)
If Awal(i, j) < 0 Then
    SelAwal(i, j) = Awal(i, j) * (-1)
Else
    SelAwal(i, j) = Awal(i, j)
End If
Next
Next

```

'Koordinat X

```

minX = SelAwal(1, 1)
For j = 1 To Val(Form1!Text(0).Text)
    If minX < SelAwal(1, j) Then
        minX = minX
    ElseIf minX = SelAwal(1, j) Then
        minX = minX
    ElseIf minX > SelAwal(1, j) Then
        minX = SelAwal(1, j)
    End If
Next

```

```

For j = 1 To Val(Form1!Text(0).Text)
    HasilX = Awal(1, j)
    If HasilX < 0 Then
        IsiX(j) = HasilX + minX
    Else
        IsiX(j) = HasilX - minX
    End If
Next

```

'Tabel X

```

Grid1.Cols = Val(Form1!Text1(1).Text) + 1
Grid1.Rows = Val(Form1!Text4(0).Text) + 1
Grid1.FixedCols = 1

Grid1.Row = 0
Grid1.Col = 0
Grid1.FixedAlignment(0) = 2: Grid1.Text = "M \ N"
Grid1.ColWidth(0) = 600

```

```

For i = 1 To Val(Form1!Text1(1).Text)
    Grid1.Col = i
    Grid1.ColWidth(i) = 1000
    Grid1.FixedAlignment(i) = 2
Next

```

n = 0

```

For k = 1 To Val(Form1!Text1(1).Text)
    Grid1.Row = 0: n = n + 1: Grid1.Col = n
    Grid1.Text = k - 1
Next k

```

```

n = 0
For i = 1 To Val(Form1!Text4(0).Text)
    Grid1.Col = 0: n = n + 1: Grid1.Row = n
    Grid1.Text = i - 1
Next

```

```

For i = 1 To Val(Form1!Text1(1).Text)
    Grid1.Col = i: Grid1.Row = 1: Grid1.ColAlignment(i) = 1
    Grid1.Text = Format(IsiX(i), "0.000")
Next

```

```

n = 1
For i = 1 To Val(Form1!Text4(0).Text) - 1
    For j = i * Val(Form1!Text1(1).Text) + 1 To i * Val(Form1!Text1(1).Text) +
Val(Form1!Text1(1).Text)
        Grid1.Col = j - i * Val(Form1!Text1(1).Text): Grid1.Row = n + i
        Grid1.Text = Format(IsiX(j), "0.000")
    Next
Next

```

```

'Koordinat Y
minY = SelAwal(2, 1)
For j = 1 To Val(Form1!Text(0).Text)
    If minY < SelAwal(2, j) Then
        minY = minY
    ElseIf minY = SelAwal(2, j) Then
        minY = minY
    ElseIf minY > SelAwal(2, j) Then
        minY = SelAwal(2, j)
    End If
Next

```

```

For j = 1 To Val(Form1!Text(0).Text)
    HasilY = Awal(2, j)
    If HasilY < 0 Then
        IsiY(j) = HasilY + minY
    Else
        IsiY(j) = HasilY - minY
    End If
Next

```

```

'Tabel Y
Grid2.Cols = Val(Form1!Text1(1).Text) + 1
Grid2.Rows = Val(Form1!Text4(0).Text) + 1

```



Next

For j = 1 To Val(Form1!Text(0).Text)

    HasilZ = Awal(3, j)

    If HasilZ < 0 Then

        IsiZ(j) = HasilZ + minZ

    Else

        IsiZ(j) = HasilZ - minZ

    End If

Next

'Tabel Z

Grid3.Cols = Val(Form1!Text1(1).Text) + 1

Grid3.Rows = Val(Form1!Text4(0).Text) + 1

Grid3.FixedCols = 1

Grid3.Row = 0

Grid3.Col = 0

Grid3.FixedAlignment(0) = 2: Grid3.Text = "M \ N"

Grid3.ColWidth(0) = 600

For i = 1 To Val(Form1!Text1(1).Text)

    Grid1.Col = i

    Grid3.FixedAlignment(i) = 2

    Grid3.ColWidth(i) = 1000

Next

n = 0

For k = 1 To Val(Form1!Text1(1).Text)

    Grid3.Row = 0: n = n + 1: Grid3.Col = n

    Grid3.Text = k - 1

Next k

n = 0

For i = 1 To Val(Form1!Text4(0).Text)

    Grid3.Col = 0: n = n + 1: Grid3.Row = n

    Grid3.Text = i - 1

Next

For i = 1 To Val(Form1!Text1(1).Text)

    Grid3.Col = i: Grid3.Row = 1: Grid3.ColAlignment(i) = 1

    Grid3.Text = Format(IsiZ(i), "0.000")

Next

n = 1

For i = 1 To Val(Form1!Text4(0).Text) - 1

    For j = i \* Val(Form1!Text1(1).Text) + 1 To i \* Val(Form1!Text1(1).Text) +  
        Val(Form1!Text1(1).Text)

        Grid3.Col = j - i \* Val(Form1!Text1(1).Text): Grid3.Row = n + i

```

Grid3.Text = Format(IsiZ(j), "0.000")
Next
Next

```

'Tabel Toleransi Potongan M

```

Grid4.Cols = Val(Form1!Text1(1).Text) + 1
Grid4.Rows = Val(Form1!Text4(0).Text) + 3
Grid4.FixedCols = 1

```

```

Grid4.Row = 0
Grid4.Col = 0
Grid4.FixedAlignment(0) = 2: Grid4.Text = "M \ N"
Grid4.ColWidth(0) = 600
For i = 1 To Val(Form1!Text1(1).Text)
    Grid4.Row = 0
    Grid4.Col = i
    Grid4.ColWidth(i) = 600: Grid4.FixedAlignment(i) = 2
Next

```

```

n = 0
For k = 1 To Val(Form1!Text1(1).Text)
    Grid4.Row = 0: n = n + 1: Grid4.Col = n
    Grid4.Text = k - 1
Next k

```

```

n = 0
For i = 1 To Val(Form1!Text4(0).Text) + 2
    Grid4.Col = 0: n = n + 1: Grid4.Row = n
    Grid4.Text = i - 1
Next

```

```

a = Val(Form1!text5.Text)
For i = 1 To Val(Form1!Text4(0).Text) + 2
    For j = 1 To Val(Form1!Text1(1).Text)
        Grid4.Col = j: Grid4.Row = i: Grid4.ColAlignment(j) = 2
        If i = (Form1!Text4(0).Text) + 1 Then
            Grid4.Text = a
            a = Val(Form1!text5.Text)
        ElseIf i = (Form1!Text4(0).Text) + 2 Then
            Grid4.Text = -a
            a = Val(Form1!text5.Text)
        End If
    Next
Next
Next

```

'Tabel Toleransi Potongan N

```

Grid5.Cols = Val(Form1!Text1(1).Text) + 3
Grid5.Rows = Val(Form1!Text4(0).Text) + 1
Grid5.FixedCols = 1

```

```

Grid5.Row = 0
Grid5.Col = 0
Grid5.FixedAlignment(0) = 2: Grid5.Text = "M \ N"
Grid5.ColWidth(0) = 600
For i = 1 To Val(Form1!Text1(1).Text)
    Grid5.Row = 0
    Grid5.Col = i
    Grid5.ColWidth(i) = 600: Grid5.FixedAlignment(i) = 2
Next

```

```

n = 0
For k = 1 To Val(Form1!Text1(1).Text) + 2
    Grid5.Row = 0: n = n + 1: Grid5.Col = n
    Grid5.Text = k - 1
Next k

```

```

n = 0
For i = 1 To Val(Form1!Text4(0).Text)
    Grid5.Col = 0: n = n + 1: Grid5.Row = n
    Grid5.Text = i - 1
Next

```

```

a = Val(Form1!text5.Text)
For i = 1 To Val(Form1!Text1(1).Text) + 2
    For j = 1 To Val(Form1!Text4(0).Text)
        Grid5.Col = i: Grid5.Row = j: Grid5.ColAlignment(i) = 2
        If i = (Form1!Text1(1).Text) + 1 Then
            Grid5.Text = a
            a = Val(Form1!text5.Text)
        ElseIf i = (Form1!Text1(1).Text) + 2 Then
            Grid5.Text = -a
            a = Val(Form1!text5.Text)
        End If
    Next
Next

```

End Sub

Private Sub Command1\_Click()

```

Screen.MousePointer = 11
Form3a.Show
Screen.MousePointer = 0

```

End Sub



```
Private Sub Command2_Click()
```

```
    Screen.MousePointer = 11  
    Form3b.Show  
    Screen.MousePointer = 0
```

```
End Sub
```

```
Private Sub Command3_Click()
```

```
    Screen.MousePointer = 11  
    Form3c.Show  
    Screen.MousePointer = 0
```

```
End Sub
```

```
Private Sub Command4_Click()
```

```
    Screen.MousePointer = 11  
    Form4a.Show  
    Screen.MousePointer = 0
```

```
End Sub
```

```
Private Sub Command5_Click()
```

```
    Screen.MousePointer = 11  
    Form4b.Show  
    Screen.MousePointer = 0
```

```
End Sub
```

```
Private Sub Command6_Click()
```

```
    Screen.MousePointer = 11  
    Form4c.Show  
    Screen.MousePointer = 0
```

```
End Sub
```

```
Private Sub Command7_Click()
```

```
    Unload Form2
```

```
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)
```

```
    Unload Form2
```

```
End Sub
```

## KURVAMX.Frm (Form 3a)

```
Private Sub Form_Load()
```

```
    Dim i As Integer
```

```
    Dim j As Integer
```

```
    Graph1.AutoInc = 1
```

```
    Graph1.NumSets = Val(Form1!Text4(0).Text) + 2
```

```
    Graph1.NumPoints = Val(Form1!Text1(1).Text)
```

```
    For i = 1 To Val(Form1!Text4(0).Text) + 2
```

```
        For j = 1 To Val(Form1!Text1(1).Text)
```

```
            If i <= Val(Form1!Text4(0).Text) Then
```

```
                Form2!Grid1.Row = i: Form2!Grid1.Col = j
```

```
                Graph1.GraphData = Val(Form2!Grid1.Text)
```

```
            Else
```

```
                Form2!Grid4.Row = i: Form2!Grid4.Col = j
```

```
                Graph1.GraphData = Val(Form2!Grid4.Text)
```

```
            End If
```

```
        Next
```

```
    Next
```

```
    For k = 0 To Val(Form1!Text4(0).Text) + 1
```

```
        If k < Val(Form1!Text4(0).Text) Then
```

```
            Graph1.LegendText = "M" & k
```

```
        ElseIf k = Val(Form1!Text4(0).Text) Then
```

```
            Graph1.LegendText = "Toleransi (+)"
```

```
        ElseIf k = Val(Form1!Text4(0).Text) + 1 Then
```

```
            Graph1.LegendText = "Toleransi (-)"
```

```
        End If
```

```
    Next
```

```
    Graph1.DrawMode = 2
```

```
End Sub
```

```
Private Sub Graph1_Click()
```

```
    Unload Form3a
```

```
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)
```

```
    Unload Form3a
```

```
End Sub
```

## KURVAMY.Frm (Form 3b)

Private Sub Form\_Load()

Dim i As Integer

Dim j As Integer

Graph1.AutoInc = 1

Graph1.NumSets = Val(Form1!Text4(0).Text) + 2

Graph1.NumPoints = Val(Form1!Text1(1).Text)

For i = 1 To Val(Form1!Text4(0).Text) + 2

For j = 1 To Val(Form1!Text1(1).Text)

If i <= Val(Form1!Text4(0).Text) Then

Form2!Grid2.Row = i: Form2!Grid2.Col = j

Graph1.GraphData = Val(Form2!Grid2.Text)

Else

Form2!Grid4.Row = i: Form2!Grid4.Col = j

Graph1.GraphData = Val(Form2!Grid4.Text)

End If

Next

Next

For k = 0 To Val(Form1!Text4(0).Text) + 1

If k < Val(Form1!Text4(0).Text) Then

Graph1.LegendText = "M" & k

Elseif k = Val(Form1!Text4(0).Text) Then

Graph1.LegendText = "Toleransi (+)"

Elseif k = Val(Form1!Text4(0).Text) + 1 Then

Graph1.LegendText = "Toleransi (-)"

End If

Next

Graph1.DrawMode = 2

End Sub

Private Sub Graph1\_Click()

Unload Form3b

End Sub

Private Sub Form\_Unload(Cancel As Integer)

Unload Form3b

End Sub





## KURVAMZ.Frm (Form 3c)

```
Private Sub Form_Load()
```

```
    Dim i As Integer
```

```
    Dim j As Integer
```

```
    Graph1.Autolnc = 1
```

```
    Graph1.NumSets = Val(Form1!Text4(0).Text) + 2
```

```
    Graph1.NumPoints = Val(Form1!Text1(1).Text)
```

```
    For i = 1 To Val(Form1!Text4(0).Text) + 2
```

```
        For j = 1 To Val(Form1!Text1(1).Text)
```

```
            If i <= Val(Form1!Text4(0).Text) Then
```

```
                Form2!Grid3.Row = i: Form2!Grid3.Col = j
```

```
                Graph1.GraphData = Val(Form2!Grid3.Text)
```

```
            Else
```

```
                Form2!Grid4.Row = i: Form2!Grid4.Col = j
```

```
                Graph1.GraphData = Val(Form2!Grid4.Text)
```

```
            End If
```

```
        Next
```

```
    Next
```

```
    For k = 0 To Val(Form1!Text4(0).Text) + 1
```

```
        If k < Val(Form1!Text4(0).Text) Then
```

```
            Graph1.LegendText = "M" & k
```

```
        ElseIf k = Val(Form1!Text4(0).Text) Then
```

```
            Graph1.LegendText = "Toleransi (+)"
```

```
        ElseIf k = Val(Form1!Text4(0).Text) + 1 Then
```

```
            Graph1.LegendText = "Toleransi (-)"
```

```
        End If
```

```
    Next
```

```
    Graph1.DrawMode = 2
```

```
End Sub
```

```
Private Sub Graph1_Click()
```

```
    Unload Form3c
```

```
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)
```

```
    Unload Form3c
```

```
End Sub
```

## KURVANX.Frm (Form 4a)

```
Private Sub Form_Load()
```

```
    Dim i As Integer
```

```
    Dim j As Integer
```

```
    Graph1.AutoInc = 1
```

```
    Graph1.NumSets = Val(Form1!Text1(1).Text) + 2
```

```
    Graph1.NumPoints = Val(Form1!Text4(0).Text)
```

```
    For i = 1 To Val(Form1!Text1(1).Text) + 2
```

```
        For j = 1 To Val(Form1!Text4(0).Text)
```

```
            If i <= Val(Form1!Text1(1).Text) Then
```

```
                Form2!Grid1.Col = i: Form2!Grid1.Row = j
```

```
                Graph1.GraphData = Val(Form2!Grid1.Text)
```

```
            Else
```

```
                Form2!Grid5.Col = i: Form2!Grid5.Row = j
```

```
                Graph1.GraphData = Val(Form2!Grid5.Text)
```

```
            End If
```

```
        Next
```

```
    Next
```

```
    For k = 0 To Val(Form1!Text1(1).Text) + 1
```

```
        If k < Val(Form1!Text1(1).Text) Then
```

```
            Graph1.LegendText = "N" & k
```

```
        ElseIf k = Val(Form1!Text1(1).Text) Then
```

```
            Graph1.LegendText = "Toleransi (+)"
```

```
        ElseIf k = Val(Form1!Text1(1).Text) + 1 Then
```

```
            Graph1.LegendText = "Toleransi (-)"
```

```
        End If
```

```
    Next
```

```
    Graph1.DrawMode = 2
```

```
End Sub
```

```
Private Sub Graph1_Click()
```

```
    Unload Form4a
```

```
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)
```

```
    Unload Form4a
```

```
End Sub
```

## KURVANY.Frm (Form 4b)

```
Private Sub Form_Load()
```

```
    Dim i As Integer
```

```
    Dim j As Integer
```

```
    Graph1.AutoInc = 1
```

```
    Graph1.NumSets = Val(Form1!Text1(1).Text) + 2
```

```
    Graph1.NumPoints = Val(Form1!Text4(0).Text)
```

```
    For i = 1 To Val(Form1!Text1(1).Text) + 2
```

```
        For j = 1 To Val(Form1!Text4(0).Text)
```

```
            If i <= Val(Form1!Text1(1).Text) Then
```

```
                Form2!Grid2.Col = i: Form2!Grid2.Row = j
```

```
                Graph1.GraphData = Val(Form2!Grid2.Text)
```

```
            Else
```

```
                Form2!Grid5.Col = i: Form2!Grid5.Row = j
```

```
                Graph1.GraphData = Val(Form2!Grid5.Text)
```

```
            End If
```

```
        Next
```

```
    Next
```

```
    For k = 0 To Val(Form1!Text1(1).Text) + 1
```

```
        If k < Val(Form1!Text1(1).Text) Then
```

```
            Graph1.LegendText = "N" & k
```

```
        ElseIf k = Val(Form1!Text1(1).Text) Then
```

```
            Graph1.LegendText = "Toleransi (+)"
```

```
        ElseIf k = Val(Form1!Text1(1).Text) + 1 Then
```

```
            Graph1.LegendText = "Toleransi (-)"
```

```
        End If
```

```
    Next
```

```
    Graph1.DrawMode = 2
```

```
End Sub
```

```
Private Sub Graph1_Click()
```

```
    Unload Form4b
```

```
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)
```

```
    Unload Form4b
```

```
End Sub
```



## KURVANZ.Frm (Form 4c)

```
Private Sub Form_Load()
```

```
    Dim i As Integer
```

```
    Dim j As Integer
```

```
    Graph1.AutoInc = 1
```

```
    Graph1.NumSets = Val(Form1!Text1(1).Text) + 2
```

```
    Graph1.NumPoints = Val(Form1!Text4(0).Text)
```

```
    For i = 1 To Val(Form1!Text1(1).Text) + 2
```

```
        For j = 1 To Val(Form1!Text4(0).Text)
```

```
            If i <= Val(Form1!Text1(1).Text) Then
```

```
                Form2!Grid3.Col = i: Form2!Grid3.Row = j
```

```
                Graph1.GraphData = Val(Form2!Grid3.Text)
```

```
            Else
```

```
                Form2!Grid5.Col = i: Form2!Grid5.Row = j
```

```
                Graph1.GraphData = Val(Form2!Grid5.Text)
```

```
            End If
```

```
        Next
```

```
    Next
```

```
    For k = 0 To Val(Form1!Text1(1).Text) + 1
```

```
        If k < Val(Form1!Text1(1).Text) Then
```

```
            Graph1.LegendText = "N" & k
```

```
        ElseIf k = Val(Form1!Text1(1).Text) Then
```

```
            Graph1.LegendText = "Toleransi (+)"
```

```
        ElseIf k = Val(Form1!Text1(1).Text) + 1 Then
```

```
            Graph1.LegendText = "Toleransi (-)"
```

```
        End If
```

```
    Next
```

```
    Graph1.DrawMode = 2
```

```
End Sub
```

```
Private Sub Graph1_Click()
```

```
    Unload Form4c
```

```
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)
```

```
    Unload Form4c
```

```
End Sub
```

## KTR.Frm (Form 5)

(Declaration)

Dim a, b, c, d, e, f As String

Private Sub Form\_Load()

Top = Screen.Height / 2 - Height / 2  
Left = Screen.Width / 2 - Width / 2

a = "- (M,N)"  
b = "- M"  
c = "- N"  
d = "- Toleransi"  
e = "- Data Design"  
f = "- Data Aktual"

Combo1.AddItem a  
Combo1.AddItem b  
Combo1.AddItem c  
Combo1.AddItem d  
Combo1.AddItem e  
Combo1.AddItem f

End Sub

Private Sub Form\_Unload(Cancel As Integer)

Unload Form5

End Sub

Private Sub Command1\_Click()

Unload Me

End Sub

Private Sub Combo1\_Click()

If Combo1.Text = a Then  
List1.Clear  
List1.AddItem "Merupakan aturan penggambaran 3DMesh."  
List1.AddItem "Urutan pengisian tabel sesuai aturan penggambaran"  
List1.AddItem "mesh."  
End If

If Combo1.Text = b Then

```
List1.Clear
List1.AddItem "Merupakan jumlah baris (M) pada gambar 3DMesh."
List1.AddItem "Nilai M antara 2 - 256."
End If
```

```
If Combo1.Text = c Then
    List1.Clear
    List1.AddItem "Merupakan jumlah kolom (N) pada gambar 3DMesh."
    List1.AddItem "Nilai N antara 2 - 256."
End If
```

```
If Combo1.Text = d Then
    List1.Clear
    List1.AddItem "Merupakan suatu nilai yang menyatakan batasan"
    List1.AddItem "besar penyimpangan dapat diterima."
End If
```

```
If Combo1.Text = e Then
    List1.Clear
    List1.AddItem "Merupakan data titik koordinat 3D yang berasal"
    List1.AddItem "dari gambar desain."
End If
```

```
If Combo1.Text = f Then
    List1.Clear
    List1.AddItem "Merupakan data titik koordinat 3D yang berasal"
    List1.AddItem "dari komponen aktual yang telah jadi."
End If
```

```
End Sub
```



## 2. AUTOLISP

```
(defun del ()  
  (command "_files" 3 "c:/prog-end/matching.dwg")  
  (command "_files" 3 "c:/prog-end/NE1.bmp")  
  (command "_files" 3 "c:/prog-end/NE2.bmp")  
  (command "_files" 3 "c:/prog-end/NW1.bmp")  
  (command "_files" 3 "c:/prog-end/NW2.bmp")  
  (command "_files" 3 "c:/prog-end/SW1.bmp")  
  (command "_files" 3 "c:/prog-end/SW2.bmp")  
  (command "_files" 3 "c:/prog-end/SE1.bmp")  
  (command "_files" 3 "c:/prog-end/SE2.bmp" "" "")  
)
```

```
(defun b ()  
  (setq tk (read-line a))  
  (command tk)  
)
```

```
(defun act ()  
  (command "3dmesh" m n)  
  (b)  
  (while (/= tk "X")  
    layer(b)  
  )  
)
```

```
(defun des ()  
  (command "layer" "m" "1" "c" "1" "" "")  
  (command "3dmesh" m n)  
  (b)  
  (while (/= tk "X")  
    (b)  
  )  
)
```

```
(defun sb()  
  (command "layer" "m" "3" "" "")  
  (command "line" "0,0,0" ox "")  
  (command "line" "0,0,0" oy "")  
  (command "line" "0,0,0" oz "")  
)
```

```
(defun txt()  
  (command "text" px t "0" "X" "")  
  (command "text" py t "0" "Y" "")  
  (command "text" pz t "0" "Z" "")  
)
```

```

(defun c:bc1 (/ m n a ox oy oz px py pz t tk)
  (del)
  (command "layer" "m" "0" "c" "5" "" "")
  (setq a (open "c:/prog-end/matching1.txt" "r"))
  (setq m (read-line a))
  (setq n (read-line a))
  (setq ox (read-line a))
  (setq oy (read-line a))
  (setq oz (read-line a))
  (setq px (read-line a))
  (setq py (read-line a))
  (setq pz (read-line a))
  (setq t (read-line a))
  (act)
  (des)
  (sb)
  (txt)
  (command "vpoint" "1,1,1")
  (command "zoom" "e" "")
  (command "zoom" "c" "0,0,0" "")
  (command "_bmpout" "c:/prog-end/NE1.bmp" "" "")
  (command "vpoint" "-1,1,1")
  (command "zoom" "e" "")
  (command "zoom" "c" "0,0,0" "")
  (command "zoom" "d" "")
  (command "_bmpout" "c:/prog-end/NW1.bmp" "" "")
  (command "vpoint" "-1,-1,1")
  (command "zoom" "e" "")
  (command "zoom" "c" "0,0,0" "")
  (command "zoom" "d" "")
  (command "_bmpout" "c:/prog-end/SW1.bmp" "" "")
  (command "vpoint" "1,-1,1")
  (command "zoom" "e" "")
  (command "zoom" "c" "0,0,0" "")
  (command "zoom" "d" "")
  (command "_bmpout" "c:/prog-end/SE1.bmp" "" "")
  (command "erase" "all" "")
)

```

```

(defun c:bc2 (/ m n a ox oy oz px py pz t tk)
  (command "layer" "m" "0" "c" "5" "" "")
  (setq a (open "c:/prog-end/matching2.txt" "r"))
  (setq m (read-line a))
  (setq n (read-line a))
  (setq ox (read-line a))
  (setq oy (read-line a))
  (setq oz (read-line a))
  (setq px (read-line a))
  (setq py (read-line a))

```